

Oracle Integrated Lights Out Manager (ILOM) 3.0

SNMP, IPMI, CIM, WS-MAN Protocol Management
Reference



Part No.: E21452-03
August 2012

Copyright © 2008, 2009, 2010, 2011, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2008, 2009, 2010, 2011, 2012, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.



Contents

Using This Documentation xi

- ▼ Download Product Software and Firmware xii

SNMP Overview 1

About Simple Network Management Protocol 1

SNMP Components 2

Oracle ILOM SNMP MIBs 3

Configuring SNMP Settings in Oracle ILOM 7

Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (CLI) 8

- ▼ Set SNMP Access and Authorization (CLI) 8

Managing SNMP User Accounts and Communities (CLI) 10

Before You Begin – SNMP User Accounts (CLI) 11

SNMP User Account Targets, Properties, and Values 11

- ▼ View and Configure SNMP Community Properties (CLI) 13

- ▼ Add an SNMP User Account (CLI) 14

- ▼ Edit an SNMP User Account (CLI) 14

- ▼ Delete an SNMP User Account (CLI) 15

- ▼ Set SNMPv3 User Account Privacy Protocol Value (CLI) 15

- ▼ Add or Edit an SNMP Community (CLI) 15

- ▼ Delete an SNMP Community (CLI) 15

Managing SNMP Trap Alerts Using the Oracle ILOM CLI 16

▼	Configure SNMP Trap Rule Destinations and Properties (CLI)	16
	CLI Commands for Managing Alert Rule Configurations	17
	Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (Web)	19
▼	Set SNMP Read and Write Access and Authorization (Web)	20
	Managing SNMP User Accounts and Communities (Web)	22
	Before You Begin - SNMP User Accounts (Web)	23
▼	Add or Edit an SNMP Community (Web)	23
▼	Delete an SNMP Community (Web)	24
▼	Add or Edit an SNMP User Account Using the Web Interface	25
▼	Delete an SNMP User Account (Web)	27
▼	Manage SNMP Trap Alerts (Web)	27
	Downloading SNMP MIBs Using Oracle ILOM	30
	Before You Begin - Download SNMP MIBs	30
▼	Download SNMP MIBs (CLI)	30
▼	Download SNMP MIBs (Web)	31
	Manage User Accounts Using SNMP	33
	Before You Begin - User Accounts (SNMP)	34
	Configuring User Accounts (SNMP)	34
▼	Configure User Accounts (SNMP)	35
▼	Configure Single Sign On (SNMP)	37
	Configuring Active Directory Settings	38
▼	Manage Active Directory Settings (SNMP)	38
▼	Manage Active Directory Administrator Groups (SNMP)	43
▼	Manage Active Directory Operator Group (SNMP)	44
▼	Manage Active Directory Custom Group (SNMP)	46
▼	Manage Active Directory User Domains (SNMP)	48
▼	Manage Active Directory Alternate Server (SNMP)	50

- ▼ Manage Server Redundancy (SNMP) 53
- ▼ Manage Active Directory DNS Locator (SNMP) 54
- ▼ Manage DNS Name Server Settings (SNMP) 56

Configuring ILOM for LDAP (SNMP) 58

- ▼ Configure LDAP Settings (SNMP) 58

Configuring ILOM for LDAP/SSL 61

- ▼ Manage LDAP/SSL Certificate (SNMP) 61
- ▼ Manage LDAP/SSL Administrator Group (SNMP) 62
- ▼ Manage LDAP/SSL Operator Group (SNMP) 63
- ▼ Manage LDAP/SSL Custom Group (SNMP) 65
- ▼ Manage LDAP/SSL User Domain (SNMP) 67
- ▼ Manage LDAP/SSL Alternate Server (SNMP) 68

Configuring RADIUS Settings (SNMP) 71

- ▼ Configure RADIUS Settings (SNM)) 71

Manage Component Information and Email Alerts (SNMP) 75

Before You Begin - Component Information (SNMP) 76

Viewing Component Information 76

- ▼ View Component Information 76

Managing Clock Settings, Event Log, Syslog Receiver, and Alert Rules 78

- ▼ View and Set Clock Settings 78
- ▼ View and Clear the ILOM Event Log 79
- ▼ Configure Remote Syslog IP Destinations 81
- ▼ Configure Severity Level Alert Rule 82

Configuring SMTP Client for Email Alert Notifications 84

- ▼ Configure SMTP Client for Alert Notification (SNMP) 84

Configuring Email Alert Settings (SNMP) 86

- ▼ Manage Email Alert Settings (SNMP) 86

Configuring Telemetry Harness Daemon (SNMP) 87

- ▼ Manage Telemetry Harness Daemon Settings (SNMP) 88

Monitor and Manage System Power (SNMP) 91

Before You Begin - Power Management (SNMP) 91

Monitoring the Power Consumption Interfaces (SNMP) 92

- ▼ Monitor System Total Power Consumption (SNMP) 93
- ▼ Monitor Actual Power Consumption (SNMP) 93
- ▼ Monitor Individual Power Supply Consumption (SNMP) 94
- ▼ Monitor Available Power (SNMP) 96
- ▼ Monitor Hardware Configuration Maximum Power Consumption (SNMP) 96
- ▼ Monitor Permitted Power Consumption (SNMP) 96
- ▼ Monitor Power Management Properties (SNMP) 96

Maintaining System Power Policy (SNMP) 97

- ▼ View and Set the Power Policy (SNMP) 98

Managing System Power Properties (SNMP) 98

- ▼ Power On System (SNMP) 99
- ▼ Reset System Power (SNMP) 99

Manage Oracle ILOM Firmware Updates (SNMP) 101

- ▼ Update Oracle ILOM Firmware (SNMP) 101

Manage ILOM Backup and Restore Configurations (SNMP) 105

- ▼ View and Configure Backup and Restore Properties (SNMP) 105

Manage SPARC Diagnostics, POST, and Boot Mode Operations (SNMP) 109

Before You Begin - Manage SPARC Hosts (SNMP) 109

Managing SPARC Diagnostic, POST, and Boot Mode Properties (SNMP) 110

- ▼ Manage SPARC Host Diagnostic Properties (SNMP) 110
- ▼ Manage SPARC Host POST Operations (SNMP) 113
- ▼ Manage SPARC Host Boot Mode Properties (SNMP) 117

- ▼ Manage SPARC Host Keyswitch Property (SNMP) 118

Server Management Using IPMI 119

Intelligent Platform Management Interface (IPMI) 119

About IPMI 120

IPMItool 121

IPMI Alerts 121

IPMI Administrator and Operator Roles 122

Configuring the IPMI State 122

- ▼ Enable IPMI State (CLI) 123

- ▼ Enable IPMI State (Web) 123

Using IPMItool to Run ILOM CLI Commands 123

Before You Begin - IPMItool and ILOM Requirements 124

- ▼ Access the ILOM CLI From IPMItool 124

Scripting ILOM CLI Commands With IPMItool 125

Performing System Management Tasks (IPMItool) 126

Before You Begin - ILOM and IPMItool Requirements 126

- ▼ Display Sensor List (IPMItool) 127

- ▼ View Single Sensor Details (IPMItool) 128

- ▼ View and Interpret Presence Sensor Type Values 128

- ▼ Power On Host (IPMItool) 130

- ▼ Power Off Host (IPMItool) 130

- ▼ Power Cycle Host (IPMItool) 130

- ▼ Shut Down Host Gracefully (IPMItool) 131

- ▼ Manage ILOM Power Budget Interfaces (IPMItool) 131

- ▼ Display FRU Manufacturing Details (IPMItool) 135

- ▼ Display ILOM Event Log Using IPMItool 136

IPMItool Utility and Command Summary 137

Server Management Using WS-Management and CIM	141
WS-Management and CIM Overview	141
WS-Management	142
Common Information Model (CIM)	142
System Management Architecture for Server Management (SMASH)	142
Configuring Support for WS-Management in ILOM	143
Before You Begin - WS-Management Requirements	143
▼ Edit the WS-Management Service State, Transport Mode, and Port Number (CLI)	143
▼ Edit WS-Management State, Transport Mode, and Port Number (Web)	146
Supported DMTF SMASH Profiles, CIM Classes and CIM Indications	147
Supported DMTF SMASH Profiles and CIM Classes	148
Supported CIM Indications	150
 Oracle's Sun-Supported CIM Classes	 153
Document Conventions For Oracle's Sun-Supported CIM Classes	154
Oracle_AssociatedIndicatorLED	154
Oracle_AssociatedSensor	156
Oracle_Chassis	157
Oracle_ComputerSystem	163
Oracle_ComputerSystemPackage	170
Oracle_Container	171
Oracle_ElementCapabilities	172
Oracle_ElementConformsToProfile	173
Oracle_EnabledLogicalElementCapabilities	174
Oracle_HWCompErrorOkIndication	177
Oracle_IndicatorLED	178

Oracle_InstCreation	187
Oracle_InstDeletion	188
Oracle_LogEntry	189
Oracle_LogManagesRecord	193
Oracle_Memory	194
Oracle_NumericSensor	199
Oracle_PhysicalAssetCapabilities	207
Oracle_PhysicalComponent	209
Oracle_PhysicalElementCapabilities	215
Oracle_PhysicalMemory	216
Oracle_PhysicalPackage	220
Oracle_Processor	227
Oracle_ProcessorChip	233
Oracle_Realizes	237
Oracle_RegisteredProfile	238
Oracle_RecordLog	241
Oracle_ReferencedProfile	246
Oracle_Sensor	247
Oracle_SpSystemComponent	253
Oracle_SystemDevice	254
Oracle_ThresholdIndication	255
Oracle_UseOfLog	261

SNMP Command Examples 263

snmpget Command	263
snmpwalk Command	264
snmpbulkwalk Command	265
snmptable Command	266
snmpset Command	269

snmptrapd Command 270

Index 273

Using This Documentation

This guide provides instructions for managing remote Oracle hardware devices using supported Oracle Integrated Lights Out Manager (ILOM) 3.0 management protocols. A list of the management protocols supported by Oracle ILOM are as follows: Simple Network Management Protocol (SNMP), Intelligent Platform Management Interface (IPMI), Web Service Management (WS-Man), and Common Information Model (CIM).

Use this guide in conjunction with other guides in the Oracle ILOM 3.0 Documentation Library. This guide is intended for technicians, system administrators, and authorized Oracle service providers, and users who have experience managing system hardware.

- [“Related Documentation” on page xii](#)
- [“Documentation Feedback” on page xii](#)
- [“Product Downloads” on page xii](#)
- [“Oracle ILOM 3.0 Firmware Version Numbering Scheme” on page xiii](#)
- [“Support and Accessibility” on page xiv](#)

Related Documentation

Documentation	Links
All Oracle products	http://www.oracle.com/documentation
Oracle Integrated Lights Out Manager (ILOM) 3.0 Documentation Library	http://www.oracle.com/pls/topic/lookup?ctx=ilom30
System management, single system management (SSM) security, and diagnostic documentation	http://www.oracle.com/technetwork/documentation/sys-mgmt-networking-190072.html
Oracle Hardware Management Pack 2.0	http://docs.oracle.com/cd/E19960-01/index.html

Note: To locate Oracle ILOM 3.1 documentation that is specific to your Sun server platform, see the Oracle ILOM section of the administration guide that is available for your server.

Documentation Feedback

Provide feedback on this documentation at:

<http://www.oracle.com/goto/docfeedback>

Product Downloads

Updates to the Oracle ILOM 3.0 firmware are available through standalone software updates that you can download from the My Oracle Support (MOS) web site for each Sun server or Sun blade chassis system. To download these software updates from the MOS web site, see the instructions that follow.

▼ Download Product Software and Firmware

1. Go to <http://support.oracle.com>.

2. Sign in to My Oracle Support.
3. At the top of the page, click the Patches and Updates tab.
4. In the Patches Search box, select Product or Family (Advanced Search).
5. In the Product? Is field, type a full or partial product name, for example Sun Fire X4470, until a list of matches appears, then select the product of interest.
6. In the Release? Is pull down list, click the Down arrow.
7. In the window that appears, click the triangle (>) by the product folder icon to display the choices, then select the release of interest.
8. In the Patches Search box, click Search.
A list of product downloads (listed as patches) appears.
9. Select the patch name of interest, for example Patch 10266805 for the Oracle ILOM and BIOS portion of the Sun Fire X4470 SW 1.1 release.
10. In the right-side pane that appears, click Download.

Oracle ILOM 3.0 Firmware Version Numbering Scheme

Oracle ILOM 3.0 has implemented a new version numbering scheme to help you identify which version of Oracle ILOM you are running on your system. The numbering scheme includes a five-field string, for example, a.b.c.d.e, where:

- a - Represents the major version of Oracle ILOM.
- b - Represents a minor version of Oracle ILOM.
- c - Represents the update version of Oracle ILOM.
- d - Represents a micro version of Oracle ILOM. Micro versions are managed per platform or group of platforms. See your platform Product Notes for details.
- e - Represents a nano version of Oracle ILOM. Nano versions are incremental iterations of a micro version.

For example, Oracle ILOM 3.1.2.1.a would designate:

- Oracle ILOM 3 as the major version
- Oracle ILOM 3.1 as a minor version
- Oracle ILOM 3.1.2 as the second update version
- Oracle ILOM 3.1.2.1 as a micro version

- Oracle ILOM 3.1.2.1.a as a nano version of 3.1.2.1

Tip – To identify the Oracle ILOM firmware version installed on your Sun server or CMM, click System Information --> Versions in the web interface, or type `version` in the command-line interface.

Support and Accessibility

Description	Links
Access electronic support through My Oracle Support	http://support.oracle.com For hearing impaired: http://www.oracle.com/accessibility/support.html
Learn about Oracle's commitment to accessibility	http://www.oracle.com/us/corporate/accessibility/index.html

SNMP Overview

Description	Links
Learn about Oracle ILOM support for SNMP	<ul style="list-style-type: none">• “About Simple Network Management Protocol” on page 1
Learn about management using SNMP	<ul style="list-style-type: none">• “SNMP Components” on page 2
Learn about the Oracle ILOM SNMP Management Information Base (MIB) files	<ul style="list-style-type: none">• “Oracle ILOM SNMP MIBs” on page 3

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, Oracle ILOM overview
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, CLI overview
- *Oracle ILOM 3.0 Daily Management Web Procedures*, web interface overview

About Simple Network Management Protocol

Oracle ILOM supports the Simple Network Management Protocol (SNMP), which is used to exchange data about network activity. SNMP is an open, industry-standard protocol technology that enables the management of networks and devices, or nodes, that are connected to the network. When using SNMP, data travels between a managed device (node) and a management station with network access. A managed device can be any device that runs SNMP, such as hosts, routers, web servers, or other servers on the network. SNMP messages are sent over IP using the User Datagram Protocol (UDP). Any management application that supports SNMP can manage your server.

For a more complete description of SNMP, see the five-part, introductory SNMP tutorial available at:

http://www.dpstele.com/layers/l2/snmp_l2_tut_part1.php

Oracle ILOM supports SNMP versions 1, 2c, and 3. Using SNMP v3 is strongly advised since SNMP v3 provides additional security, authentication, and privacy beyond SNMP v1 and v2c.

SNMP is a protocol, not an application, so you need an application to utilize SNMP messages. Your SNMP management software might provide this functionality, or you can use an open-source tool like Net-SNMP, which is available at:

<http://net-snmp.sourceforge.net/>

Note – Oracle ILOM users reading this document are assumed to have a working knowledge of SNMP. SNMP client-side commands are used in this text as examples of using SNMP. Users who do not have a working knowledge of SNMP should complete the tutorial at

http://net-snmp.sourceforge.net/wiki/index.php/Main_Page. This tutorial is more advanced than the introductory tutorial referred to above.

SNMP Components

SNMP functionality requires the following two components:

- **Network management station** – A network management station hosts management applications, which monitor and control managed nodes.
- **Managed node** – A managed node is a device such as a server, router, or hub that hosts SNMP management agents that are responsible for carrying out requests from management stations, such as a service processor (SP) running Oracle ILOM. Managed nodes can also provide unsolicited status information to a management station in the form of a trap.

SNMP is the protocol used to communicate management information between management stations and SNMP agents.

The SNMP agent is preinstalled on your Oracle Sun server platform and runs on Oracle ILOM, so all SNMP management occurs through Oracle ILOM. To utilize this feature, your operating system must have an SNMP client application.

Both management stations and agents use SNMP messages to communicate. Management stations can send and receive information. Agents can respond to requests and send unsolicited messages in the form of traps. Management stations and agents use the following functions:

- Get
- GetNext
- GetResponse
- Set
- Trap

Oracle ILOM SNMP MIBs

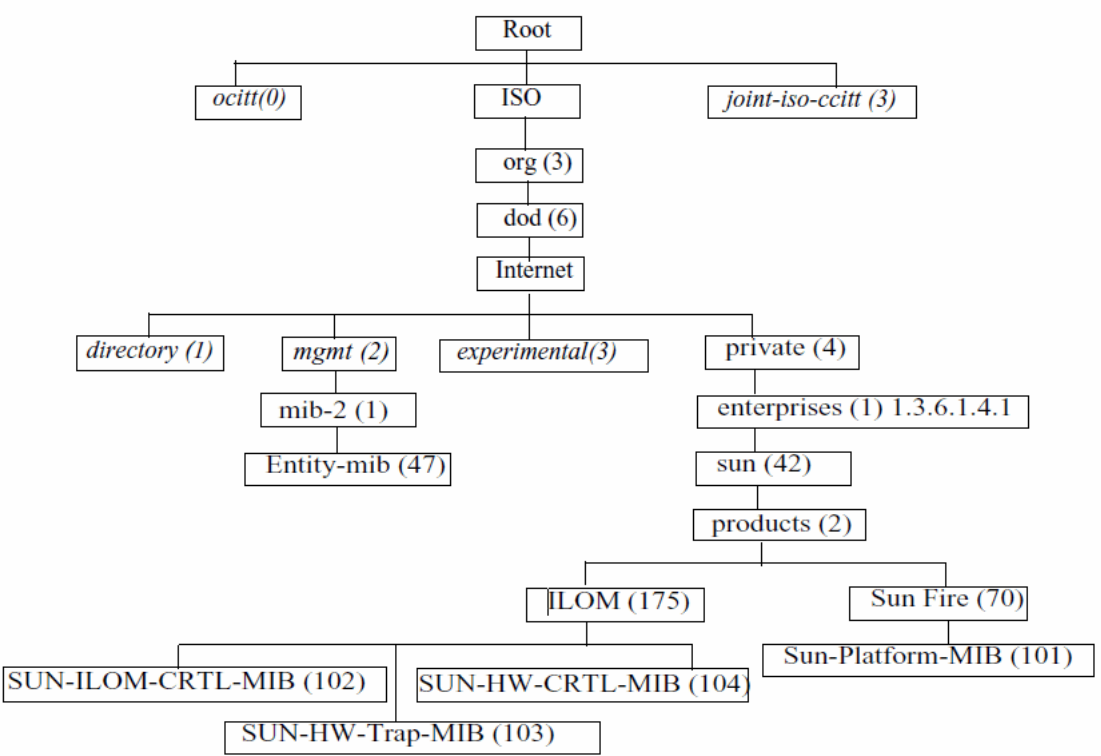
The base component of an SNMP implementation is the Management Information Base (MIB). A MIB is a text file that describes a managed node's available information. This tree-like, hierarchical system classifies information about resources in a network as a list of data objects, each with a unique identifier, or object ID. Thus, the MIB defines the data objects, or variables, that the SNMP agent can access. When a management station requests information from a managed node, the agent receives the request and retrieves the appropriate information from the MIBs. In Oracle ILOM, the MIB makes it possible to access the server's network configuration, status, and statistics.

As of Oracle ILOM 3.0.4, SNMP MIBs are a part of the Oracle ILOM firmware. You can download MIBs directly from Oracle ILOM. For more information about MIBs, and instructions for downloading MIBs from Oracle ILOM, see the following guides:

- *Oracle ILOM 3.0 Daily Management Concepts Guide*
- *Oracle ILOM 3.0 Daily Management CLI Procedures Guide*
- *Oracle ILOM 3.0 Daily Management Web Interface Procedures Guide*

The following figure shows the standard MIB tree and the location of the Oracle ILOM MIB modules in that tree. The Oracle ILOM MIB modules are described in the table that follows.

EXAMPLE: Location of Oracle ILOM MIB Modules



The following table provides a description of the Oracle ILOM MIB modules and lists the object ID for each MIB name.

TABLE: Description of Oracle ILOM MIB Modules, Object ID, and MIB Name

MIB Name	Description	MIB Object ID
ENTITY-MIB	The MIB module for representing multiple physical entities supported by a single SNMP agent. Note - The entPhysicalTable is the only part of this MIB that is implemented.	1.3.6.1.2.1.47
SUN-HW-CTRL-MIB	This MIB allows controls for all Oracle Sun server platform devices using Oracle ILOM. Note - Only the power management portions of this MIB are implemented.	1.3.6.1.4.1.42.2.175.104

TABLE: Description of Oracle ILOM MIB Modules, Object ID, and MIB Name

MIB Name	Description	MIB Object ID
SUN-HW-TRAP-MIB	This MIB describes the hardware-related notifications and traps that can be generated by Oracle Sun server platforms.	1.3.6.1.4.1.42.2.175.103
SUN-ILOM-CONTROL-MIB	This MIB provides objects for configuring and managing all Oracle ILOM functions. Configuration covered by this MIB includes functions such as authorization, authentication, logging, services, networking, and firmware management.	1.3.6.1.4.1.42.2.175.102
SUN-PLATFORM-MIB	This MIB provides extensions to the ENTITY-MIB (RFC 2737) where each entity modeled in the system is represented by means of extensions to the entPhysicalTable.	1.3.6.1.4.1.42.2.70.101

Portions of the standard MIBs listed in the following table are implemented by Oracle ILOM.

TABLE: Standard MIBs Implemented by Oracle ILOM

MIB Name	Description	MIB Object ID
IF-MIB	The MIB module for describing generic objects for network interface sub-layers. This MIB is an updated version of MIB-II's ifTable, and incorporates the extensions defined in RFC 1229.	1.3.6.1.2.1.31
IP-MIB	The MIB module for managing IP and ICMP implementations, but excluding their management of IP routes.	1.3.6.1.2.1.4.
SNMP-FRAMEWORK-MIB	The SNMP Management Architecture MIB.	1.3.6.1.6.3.10
SNMPv2-MIB	The MIB module for SNMP entities. Note - Only the system and SNMP groups from this MIB module apply to Oracle ILOM.	1.3.6.1.6.3.1
TCP-MIB	The MIB module for managing TCP implementations.	1.3.6.1.2.1.49
UDP-MIB	The MIB module for managing UDP implementations.	1.3.6.1.2.1.50

The following table describes MIBs that are used in support of the Oracle ILOM SNMP implementation.

TABLE: MIBs Used in Support of the Oracle ILOM SNMP Implementation

MIB Name	Description	MIB Object ID
HOST-RESOURC ES-MIB	This MIB is for use in managing host systems. The MIB supports attributes common to all Internet hosts including, for example, both personal computers and systems that run variants of UNIX.	1.3.6.1.2.1.25.1
IANAifType-MIB	This MIB module defines the IANAifType Textual Convention, and thus the enumerated values of the ifType object defined in MIB-II's ifTable.	1.3.6.1.2.1.30
NOTIFICATION- LOG-MIB	This MIB module is used for logging SNMP notifications (traps).	1.3.6.2.1.92.1.1.3
SNMP-MPD-MIB	This MIB module is used for message processing and dispatching.	1.3.6.1.6.3.11
SNMPv2-TM	This MIB module is used for SNMP transport mappings.	1.3.6.1.6.3.19
SNMPv2-SMI	This MIB module contains definitions for the structure of management information, version 2.	1.3.6.1.6

Configuring SNMP Settings in Oracle ILOM

Description	Links
Oracle ILOM CLI procedures for managing SNMP access, user accounts, and SNMP trap alerts	<ul style="list-style-type: none">• “Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (CLI)” on page 8• “Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (Web)” on page 19
Download SNMP MIBs directly from Oracle ILOM	<ul style="list-style-type: none">• “Downloading SNMP MIBs Using Oracle ILOM” on page 30

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, user account management
- *Oracle ILOM 3.0 Daily Management Web Procedures*, managing user accounts
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, managing user accounts

Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (CLI)

Description	Links
Learn about the requirements for SNMP management using the Oracle ILOM CLI	<ul style="list-style-type: none">• “Before You Begin – SNMP User Accounts (CLI)” on page 11
Oracle ILOM CLI procedure to enable SNMP	<ul style="list-style-type: none">• “Set SNMP Access and Authorization (CLI)” on page 8
Oracle ILOM CLI procedures for managing SNMP user account properties	<ul style="list-style-type: none">• “Managing SNMP User Accounts and Communities (CLI)” on page 10
Oracle ILOM CLI procedures for managing SNMP traps	<ul style="list-style-type: none">• “Managing SNMP Trap Alerts Using the Oracle ILOM CLI” on page 16

▼ Set SNMP Access and Authorization (CLI)

Before You Begin

- To modify SNMP properties in Oracle ILOM, you must have the Admin role (a) enabled.
- The SNMP `servicestate` property is, by default, shipped from the factory *enabled*.
- The SNMP `sets write access` property is, by default, shipped from the factory *disabled*. To allow SNMP write access to ILOM, you must enable the SNMP `sets` property.

Note – When the “Set Requests” state is disabled in Oracle ILOM, all SNMP objects are read-only and no `snmpset` commands are processed.

- Oracle ILOM provides authentication properties for each of the following SNMP protocol versions: v1, v2c, and v3.
 - For SNMP v1 and v2c, Oracle ILOM provides a `communities` property with values of *public* and *private* to manage user authentication. However, the `communities` property value for SNMPv1 and v2c are, by default, shipped from the factory *disabled*.

- For SNMP v3, Oracle ILOM provides a `users` property to manage user authentication. The `users` property is, by default, shipped from the factory *enabled*. The SNMPv3 `users` property is not shipped from the factory with pre-packaged values for users.

To set the SNMP service state, properties, follow these steps:

1. Log in to the Oracle ILOM SP CLI.

2. To view the Oracle ILOM SNMP properties, type:

-> **show /SP/services/snmp**

The following SNMP output appears.

```
-> show /SP/services/snmp
/SP/services/snmp
Targets:
  communities
  mibs
  users
Properties:
  engineid = none
  port = 161
  servicestate = (enabled)
  sets = disabled
  v1 = disabled
  v2c = disabled
  v3 = enabled
Commands:
  cd
  set
  show
```

3. Use the `set` command to change any of the SNMP properties, for example:

To enable:	Type:
SNMP with read-only access	-> set /SP/services/snmp servicestate=enabled
SNMP write access	-> set /SP/services/snmp sets=enabled
SNMP protocol version (v1, v2c, or v3) property	-> set /SP/services/snmp v#=enabled where # = the SNMP protocol version you want to enable
To create an SNMP v3 :	Type:
User account for authorizaition and provide read and write access	-> create /SP/services/snmp/users/<useraccountname> authenticationpassword=password permission=rw
User account for authorizaition and provide read only access	-> create /SP/services/snmp/users/<useraccountname> authenticationpassword=password

For more information about SNMP user accounts and read and write access, see [“Managing SNMP User Accounts and Communities \(CLI\)” on page 10.](#)

Managing SNMP User Accounts and Communities (CLI)

Topic Description	Links
Identify requirements for managing SNMP user accounts.	<ul style="list-style-type: none"> • “Before You Begin – SNMP User Accounts (CLI)” on page 11
Identify CLI targets and properties for SNMP user accounts	<ul style="list-style-type: none"> • “SNMP User Account Targets, Properties, and Values” on page 11
Procedures for configuring SNMP user accounts using the Oracle ILOM CLI	<ul style="list-style-type: none"> • “View and Configure SNMP Community Properties (CLI)” on page 13 • “Add an SNMP User Account (CLI)” on page 14 • “Edit an SNMP User Account (CLI)” on page 14 • “Set SNMPv3 User Account Privacy Protocol Value (CLI)” on page 15 • “Delete an SNMP User Account (CLI)” on page 15 • “Add or Edit an SNMP Community (CLI)” on page 15 • “Delete an SNMP Community (CLI)” on page 15

Before You Begin – SNMP User Accounts (CLI)

Prior to performing the procedures in this section, you must ensure that the following requirements are met:

- To set user account CLI properties in Oracle ILOM, you need the User Management (u) role enabled.
- Verify the proper SNMP settings are enabled in Oracle ILOM. For more details, see [“Set SNMP Access and Authorization \(CLI\)” on page 8](#).

Note – When you are working in the Oracle ILOM CLI, if the `Sets` parameter is disabled, all SNMP MIB objects are read-only.

- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user account with read-write (rw) privileges.

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work as presented only if you have Net-SNMP and the Net-SNMP sample applications installed.

SNMP User Account Targets, Properties, and Values

The SNMP user account targets, properties, and values are accessible under the `/SP/services/snmp` target. The following table identifies the targets, properties, and values that are valid for SNMP user accounts.

TABLE: SNMP User Account Targets, Properties and Values

Target	Property	Value	Default
/SP/services/snmp/ communities/ communityname	permissions	ro rw	ro
/SP/services/snmp/users/ username	authenticationprotocol	MD5 SHA	MD5
	authenticationpassword*	<string>	(null string)
	permissions	ro rw	ro
	privacyprotocol	none DES AES†	none
	privacypassword‡	<string>	(null string)
/SP/services/snmp	engineid = none	<string>	(null string)
	port = 161	<integer>	161
	servicestate = enabled	enable disabled	enabled
	sets = enabled	enabled disabled	disabled
	v1 = disabled	enabled disabled	disabled
	v2c = disabled	enabled disabled	disabled
	v3 = disabled	enabled disabled	enabled

* An authentication password must be provided when you create or modify users (SNMP v3 only).

† If the privacyprotocol property has a value other than none, then a privacy password must be set.

‡ AES (Advanced Encryption Standard) privacy protocol option is available for SNMPv 3 as of ILOM 3.0.16.

For example, to change privacyprotocol for user a1 to DES, use the following syntax:

```
-> set /SP/services/snmp/users/a1 privacyprotocol=DES
privacypassword=password authenticationprotocol=SHA
authenticationpassword=password
```

Note that the changes would be invalid if the following syntax was specified:

```
-> set /SP/services/snmp/users/a1 privacyprotocol=DES
```

Note – You can change SNMP user permissions without resetting the privacy and authentication properties.

▼ View and Configure SNMP Community Properties (CLI)

1. To go to the `/SP/services/snmp` directory, type:

```
-> cd /SP/services/snmp
```

2. Within that directory, type the `show` command to view SNMP settings. The default settings are as follows:

```
-> show
    /SP/services/snmp
Targets:
    communities
    mibs
    users
Properties:
    engineid = (none)
    port = 161
    servicestate = enabled
    sets = disabled
    v1 = disabled
    v2c = disabled
    v3 = enabled
Commands:
    cd
    set
    show
```

3. To view the communities, type:

```
-> show /SP/services/snmp/communities
```

```
-> show /SP/services/snmp/communities
/SP/services/snmp/communities
Targets:
    private
    public
Properties:
Commands:
    cd
    create
    delete
    show
```

4. To create a community with read/write privileges, type:

```
-> create /SP/services/snmp/communities/communityname  
permission=rw
```

```
-> create /SP/services/snmp/communities/communityname permission=  
rw  
Created /SP/services/snmp/communities/communityname
```

5. View the public communities by typing:

```
-> show /SP/services/snmp/communities/public
```

```
-> show /SP/services/snmp/communities/public  
/SP/services/snmp/communities/public  
Targets:  
Properties:  
    permission = ro  
Commands:  
    cd  
    set  
    show
```

▼ Add an SNMP User Account (CLI)

1. Log in to the Oracle ILOM CLI.
2. To add an SNMP v3 read-only user account, type the following command:

```
-> create /SP/services/snmp/users/username  
authenticationpassword=password
```

▼ Edit an SNMP User Account (CLI)

1. Log in to the Oracle ILOM CLI.
2. To edit an SNMP v3 user account, type the following command:

```
-> set /SP/services/snmp/users/username authenticationpassword=  
password
```

Note – When changing the parameters of SNMP users, you must provide a value for authenticationpassword, even if you are not changing the password.

▼ Delete an SNMP User Account (CLI)

1. Log in to the Oracle ILOM CLI.
2. To delete an SNMP v3 user account, type the following command:

```
-> delete /SP/services/snmp/users/username
```

▼ Set SNMPv3 User Account Privacy Protocol Value (CLI)

Before You Begin

- An SNMP user account must be created before you set a Privacy Protocol property value for the user account. For details, see [“Add an SNMP User Account \(CLI\)” on page 14](#).

1. Log in to the Oracle ILOM CLI.
2. To modify the `privacyprotocol` property value assigned to an SNMP v3 user account, type the following command:

```
-> set /SP/services/snmp/users/username privacyprotocol=  
<DES|AES|None>
```

Note – The SNMPv3 AES (Advanced Encryption Standard) option is available in Oracle ILOM as of 3.0.16.

▼ Add or Edit an SNMP Community (CLI)

1. Log in to the Oracle ILOM CLI.
2. To add an SNMP v1/v2c community, type the following command:

```
-> create /SP/services/snmp/communities/communityname
```

▼ Delete an SNMP Community (CLI)

1. Log in to the Oracle ILOM CLI.
2. To delete an SNMP v1/v2c community, type the following command:

```
-> delete /SP/services/snmp/communities/communityname
```

Managing SNMP Trap Alerts Using the Oracle ILOM CLI

Topic Descriptions	Links
CLI SNMP trap procedure	<ul style="list-style-type: none">• “Configure SNMP Trap Rule Destinations and Properties (CLI)” on page 16
CLI alert rule command reference	<ul style="list-style-type: none">• “CLI Commands for Managing Alert Rule Configurations” on page 17

▼ Configure SNMP Trap Rule Destinations and Properties (CLI)

Before You Begin

- To create or edit alert rules in Oracle ILOM, you need the Admin (a) role enabled.
- For you to define an SNMP v3 trap alert, the SNMPv3 user name must be defined in Oracle ILOM. If the SNMP v3 user name is not defined in Oracle ILOM, the SNMP v3 user receiving the SNMP alert will not be able to decode the SNMPv3 alert message. For more information about defining SNMPv3 authorization and SNMP v3 users in Oracle ILOM, see [“Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts \(CLI\)” on page 8](#).
- Review [“CLI Commands for Managing Alert Rule Configurations” on page 17](#).
- For additional information about configuring alert management settings in Oracle ILOM, refer to [“Managing System Alerts”](#) in the *Oracle ILOM 3.0 Daily Management – CLI Procedures Guide* or the *Oracle ILOM 3.0 Daily Management – Concepts Guide*.

To configure the destinations to which the SNMP traps are sent, follow these steps:

1. **Log in to the Oracle ILOM CLI.**
2. **To display the current settings of the alert rule, type the `show` command.**

For example:

```
-> show /SP/alertmgmt/rules/1
/SP/alertmgmt/rules/1
Targets:
Properties:
  community_or_username = public
  destination = 0.0.0.0
  level = disable
  snmp_version = 1
  type = snmptrap
Commands:
```

```
cd
set
show
```

3. To show the `/SP/alertmgmt/rules` directory, type these commands:

```
-> cd /SP/alertmgmt/rules
-> show
```

```
-> cd /SP/alertmgmt/rules
-> show
/SP/alertmgmt/rules
Targets:
  1
  2
  .
  .
  .
 15
Properties:
  testalert = (Cannot show property)
Commands:
  cd
  set
  show
```

Choose a rule (from targets 1 through 15) for which you would like to configure a destination for SNMP traps, and go to that directory.

For example:

```
-> cd 4
```

4. To change the rule properties, within that rule directory, type the `set` command.

For example:

```
-> set type=snmptrap level=critical destination=
IPaddress_of_snmp_management_station snmp_version=2c
community_or_username=public
```

CLI Commands for Managing Alert Rule Configurations

The following table describes the CLI commands that you will need to use to manage alert rule configurations in the Oracle ILOM CLI.

TABLE: CLI Commands for Managing Alert Rule Configurations

CLI Command	Description
show	<p>The show command enables you to display any level of the alert management command tree by specifying either the full or relative path. Examples:</p> <ul style="list-style-type: none">• To display an alert rule along with its properties using a full path, you would type the following at the command prompt: -> show /SP/alertmgmt/rules/1 /SP/alertmgmt/rules/1 Properties: community_or_username = public destination = 129.148.185.52 level = minor snmp_version = 1 type = snmptrap Commands: cd set show• To display a single property using the full path, you would type the following at the command prompt: -> show /SP/alertmgmt/rules/1 type /SP/alertmgmt/rules/1 Properties: type = snmptrap Commands: set show• To specify a relative path if the current tree location is /SP/alertmgmt/rules, you would type the following at the command prompt: -> show 1/ /SP/alertmgmt/rules/1 Targets: Properties: community_or_username = public destination = 129.148.185.52 level = minor snmp_version = 1 type = snmptrap Commands: cd set show

TABLE: CLI Commands for Managing Alert Rule Configurations *(Continued)*

CLI Command	Description
cd	The <code>cd</code> command enables you to set the working directory. To set alert management as a working directory on a server SP, you would type the following command at the command prompt: -> <code>cd /SP/alertmgmt</code>
set	The <code>set</code> command enables you to set values to properties from any place in the tree. You can specify either a full or relative path for the property depending on the location of the tree. For example: <ul style="list-style-type: none">• For full paths, you would type the following at the command prompt: -> <code>set /SP/alertmgmt/rules/1 type=snmptrap</code>• For relative path (tree location is <code>/SP/alertmgmt</code>), you would type the following command path at the command prompt: -> <code>set rules/1 type=snmptrap</code>• For relative path (tree location is <code>/SP/alertmgmt/rules/1</code>), you would type the following command path at the command prompt: -> <code>set type=snmptrap</code>



Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts (Web)

Description	Links
Web procedure for setting the SNMP service state properties in Oracle ILOM	<ul style="list-style-type: none">• “Set SNMP Read and Write Access and Authorization (Web)” on page 20
Web procedure for managing SNMP user accounts and communities	<ul style="list-style-type: none">• “Managing SNMP User Accounts and Communities (Web)” on page 22
Web procedure to managing SNMP trap properties	<ul style="list-style-type: none">• “Manage SNMP Trap Alerts (Web)” on page 27

▼ Set SNMP Read and Write Access and Authorization (Web)

Before You Begin

- To modify SNMP properties in Oracle ILOM, you must have the Admin role (a) enabled.
- The SNMP `service` state is, by default, shipped from the factory *enabled*.
- The SNMP `set requests` state is, by default, shipped from the factory *disabled*. To allow SNMP write access to ILOM, you must enable the set requests state.

Note – When the set requests state is disabled in Oracle ILOM, all SNMP objects are read-only and no `snmpset` commands are processed.

- Oracle ILOM provides authentication properties for each of the following SNMP protocol versions: v1, v2c, and v3.
 - For SNMP v1 and v2c, Oracle ILOM provides a `communities` property with values of *public* and *private* to manage user authentication. However, the property values for SNMP v1 and v2c communities are, by default, shipped from the factory *disabled*.
 - For SNMP v3, Oracle ILOM provides a `users` property to manage user authentication. The `users` property is, by default, shipped from the factory *enabled*. The SNMP v3 `users` property is not shipped from the factory with pre-packaged values for users.

To set the SNMP service state, properties, follow these steps:

1. **Log in to the Oracle ILOM web interface.**
2. **Click Configuration --> System Management Access --> SNMP.**
The SNMP Settings page appears.

System Information **System Monitoring** **Configuration** **User Management** **Remote Control** **Maintenance**

System Management Access **Alert Management** **Network** DNS Serial Port Clock Timezone

Web Server SSL Certificate **SNMP** SSH Server IPMI

SNMP Settings

Manage SNMP users, communities, and access from this page. Use the checkboxes to control the state of the SNMP Agent, permission for Set Requests, and access for each of the protocols.

State: ☒ Enabled

Port:

Engine ID:

Set Requests: ☒ Enabled

Protocols: ☒ v1 ☒ v2c ☒ v3

[Communities](#) [Users](#)

3. To enable the SNMP port, click the State check box.

When State is disabled, the SNMP port is blocked, prohibiting all SNMP communication between Oracle ILOM and the network.

4. In the Port text field, type the port number.

5. Leave the Engine ID field blank. This allows the default setting to be used.

The engine ID is automatically set by the SNMP agent. While you can use this field to set the engine ID, you should leave this field blank. The engine ID uniquely identifies the SNMP engine and enables users to query the SNMP agent. You should use this field to set the engine ID only if you are familiar with SNMP v3 security and how this setting is used.

6. To enable or disable the Set Requests option, select or clear the Set Requests check box.

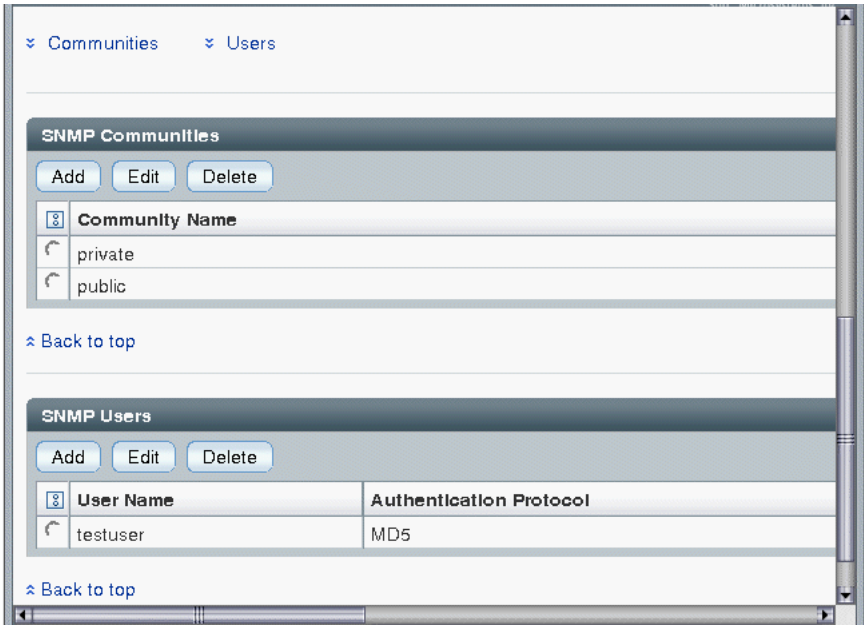
If the Set Requests option is disabled, all SNMP objects are read-only and no snmpset commands will be processed.

7. To enable SNMP v1, v2c, or v3, click a Protocols check box.

SNMP v3 is enabled by default. You can enable or disable v1, v2c, and v3 protocol versions.

8. Click Save.

At the bottom of the SNMP Settings page, you can also add, edit, or delete SNMP communities or users, as shown in the following figure.



Managing SNMP User Accounts and Communities (Web)

Description	Links
Learn about what is required before managing SNMP user accounts	<ul style="list-style-type: none">• “Before You Begin – SNMP User Accounts (CLI)” on page 11
Web procedures for configuring SNMP user accounts and communities	<ul style="list-style-type: none">• “Add or Edit an SNMP Community (Web)” on page 23• “Delete an SNMP Community (Web)” on page 24• “Add or Edit an SNMP Community (Web)” on page 23• “Delete an SNMP User Account (Web)” on page 27

Before You Begin - SNMP User Accounts (Web)

Prior to performing the procedures in this section, you must ensure that the following requirements are met:

- To set user account CLI properties in Oracle ILOM, you need the User Management (u) role enabled.
- Verify that the proper SNMP settings are enabled in Oracle ILOM. For more details, see [“Set SNMP Read and Write Access and Authorization \(Web\)” on page 20](#).

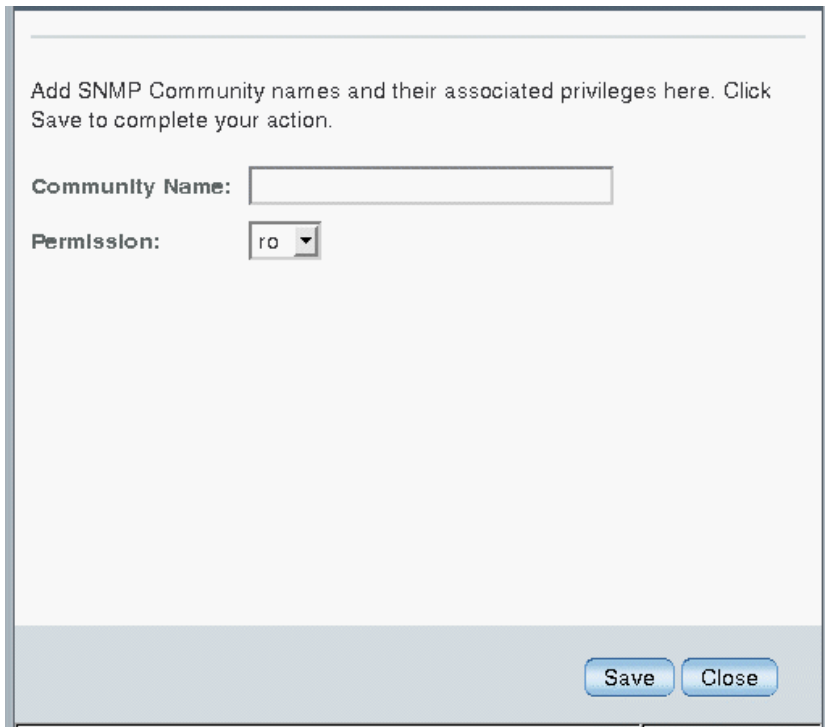
Note – When you are working in the Oracle ILOM CLI, if the `Sets` parameter is disabled, all SNMP MIB objects are read-only.

- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user account with read-write (rw) privileges.

▼ Add or Edit an SNMP Community (Web)

To add or edit an SNMP v1 or v2c community, follow these steps:

1. **Log in to the Oracle ILOM web interface.**
2. **Click Configuration --> System Management Access --> SNMP.**
Scroll to the bottom half of the SNMP Settings page to find the SNMP Communities dialog box.
3. **To add a community, click Add.**
The SNMP Community dialog box appears.

A screenshot of a web-based dialog box for configuring SNMP communities. The dialog has a light gray background and a thin border. At the top, it contains the text: "Add SNMP Community names and their associated privileges here. Click Save to complete your action." Below this text are two input fields. The first is labeled "Community Name:" and is an empty text box. The second is labeled "Permission:" and is a dropdown menu with "ro" selected. At the bottom right of the dialog, there are two buttons: "Save" and "Close".

Add SNMP Community names and their associated privileges here. Click Save to complete your action.

Community Name:

Permission:

Save Close

4. To edit a community, do the following:

- a. Click the appropriate community radio button**
- b. Click Edit.**

The SNMP Community dialog box appears.

5. If you are adding a new community, type the name of the community in the Community Name field; otherwise, proceed to the next step.

The community name can contain up to 35 characters. It must start with an alphabetic character and cannot contain a space.

6. In the Permissions drop-down list, select read-only (ro) or read-write (rw).

7. Click Save.

▼ Delete an SNMP Community (Web)

To delete an SNMP v1 or v2c community, follow these steps:

1. Log in to the Oracle ILOM web interface.

2. Click **Configuration --> System Management Access --> SNMP**.
The SNMP settings page appears.
3. Click the **Communities** link or scroll down to the communities list.
4. Click the radio button of the SNMP community to delete.
5. Click **Delete**.
A confirmation dialog box appears.
6. Click **OK** to delete the SNMP community.

▼ Add or Edit an SNMP User Account Using the Web Interface

To add or edit an SNMP v3 user account, follow these steps:

Note – User accounts are not applicable to SNMP v1 and v2c because communities are used to control access.

1. Log in to the Oracle ILOM web interface.
2. Click **Configuration --> System Management Access --> SNMP**.
The SNMP Settings page appears.
3. Click the **Users** link to expand the SNMP Settings page and display SNMP Users.
4. To add an SNMP user, click **Add**.
The Add or Edit SNMP User dialog box appears.
5. To edit an SNMP user, do the following:
 - a. Click the appropriate user radio button
 - b. Click **Edit**.
The Edit SNMP User Information dialog box appears.

Oracle® Integrated Lights Out Manager

Edit SNMP user information here. Click Save to confirm your changes.

User Name: davidc

Authentication Protocol: MD5

Authentication Password:

Confirm Password:

Permission: Read-Only

Privacy Protocol: none

Privacy Password:

Confirm Password:

Save Close

Done

6. If you are adding a user, type a user name in the User Name text field; otherwise proceed to the next step.

The user name can include up to 35 characters. It must start with an alphabetic character and cannot contain spaces.

7. In the Authentication Protocol drop-down list, select either Message Digest 5 (MD5) or Secure Hash Algorithm (SHA).

8. In the Authentication Password text field, type a password.

The authentication password is case-sensitive and must contain 8 to 16 characters, with no colons or space characters.

9. In the Confirm Password text field, retype the authentication password.

10. In the Permissions drop-down list, select read-only (ro) or read-write (rw).

11. (Optional) To specify a privacy protocol, perform the following steps:

- a. In the Privacy Protocol list box, select DES or AES.

Note – The AES (Advanced Encryption Standard) privacy protocol option is available only for SNMPv3 as of ILOM 3.0.16.

- b. In the Privacy Password text box, type a password for the privacy algorithm specified in Step 11a.

The privacy password is case-sensitive and must contain 8 to 16 characters, with no colons or space characters.

Note – The privacy password is only required if you selected DES or AES in Step 11a.

- c. In the Confirm Password field, retype the privacy password to ensure that it matches the privacy password specified in Step 11b.

12. Click **Save** to apply the SNMP user account properties.

▼ Delete an SNMP User Account (Web)

To delete an SNMP v3 user account, follow these steps:

1. Log in to the Oracle ILOM web interface.
2. Click **Configuration --> System Management Access --> SNMP**.
The SNMP Settings page appears.
3. Click the **Users** link or scroll down to the **SNMP Users** list.
4. Click the radio button of the SNMP user account to delete.
5. Click **Delete** under the **SNMP User's List**.
A confirmation dialog box opens.
6. Click **OK** to delete the user account.

▼ Manage SNMP Trap Alerts (Web)

Before You Begin

- To create or edit SNMP trap alert rules in Oracle ILOM, you need the Admin (a) role enabled.
- To define an SNMP v3 trap alert, you must define the SNMP v3 user name must be defined in Oracle ILOM. If the SNMP v3 user name is not defined in Oracle ILOM, the SNMP v3 user receiving the SNMP alert will not be able to decode the SNMP v3 alert message. For more information about defining SNMP v3 authorization and SNMP v3 users in Oracle ILOM, see [“Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts \(Web\)”](#) on page 19.
- For additional information about configuring alert management settings in Oracle ILOM, refer to “Managing System Alerts” in the *Oracle ILOM 3.0 Daily Management – Web Procedures Guide* or the *Oracle ILOM 3.0 Daily Management – Concepts Guide*.

To configure SNMP Trap Alert properties in Oracle ILOM, follow these steps:

1. **Log in to the Oracle ILOM web interface.**
2. **Click Configuration --> Alert Management.**

The Alert Settings page appears. This page shows a table of the alerts that you can configure. You can configure up to 15 alerts.

Alerts		
— Actions —		
Alert ID	Level	Alert Type
1	disable	ipmipet
2	disable	ipmipet

3. **To create or modify an alert, click the alert radio button.**
4. **From the Actions drop-down list, select Edit.**

The Create or Modify Alert dialog appears.

To create or modify an Alert, select the alert level and type, then fill in the destination information for the alert type selected.

Level:

Type:

Fill in the IP address of the PET destination. Click Save to complete your action.

IP Address:

5. In the Level drop-down list, select the level of the alert.
6. In the Type drop-down list, select the alert type.
7. In the IP Address field, specify the alert destination IP address.
8. Click Save for your changes to take effect.

Downloading SNMP MIBs Using Oracle ILOM

Description	Links	Platform Feature Support
Identify requirements for downloading SNMP MIBs from Oracle ILOM	<ul style="list-style-type: none">• “Before You Begin - Download SNMP MIBs” on page 30	<ul style="list-style-type: none">• x86 system server SP• SPARC system server SP• CMM
Download SNMP MIBs directly from Oracle ILOM CLI	<ul style="list-style-type: none">• “Download SNMP MIBs (CLI)” on page 30	
Download SNMP MIBs directly from Oracle ILOM web interface	<ul style="list-style-type: none">• “Download SNMP MIBs (Web)” on page 31	

Before You Begin - Download SNMP MIBs

- The Reset and Host Control (r) role is required for you to download SNMP MIBs from Oracle ILOM.
- You must be using Oracle ILOM 3.0.4 or a later version of Oracle ILOM.

▼ Download SNMP MIBs (CLI)

1. Log in to the Oracle ILOM CLI SP or CMM.
2. Use the `show` command to display the SNMP MIBs.

For example:

```
-> show /SP/services/snmp/mibs

/SP/services/snmp/mibs
Targets:

Properties:
  dump_uri = (Cannot show property)

Commands:
  cd
```

dump set show

3. To download the files, type either of the following commands:

-> **dump -destination** *URI* **/SP/services/snmp/mibs**

or

-> **set /SP/services/snmp/mibs dump_uri=***URI*

where *URI* specifies the target to which the files are downloaded.

A zip file containing the MIBs are transferred to the destination server.

▼ Download SNMP MIBs (Web)

1. Log in to the Oracle ILOM SP or CMM web interface.

2. Click Configuration --> System Management Access --> SNMP.

The SNMP Management page appears.

3. Click the MIBs jump link, or scroll down to the MIBs section.

The MIBs section of the page appears.

4. Click Download, then click Save and enter the destination to save the file.

A zip file containing the MIBs is transferred to the destination server.

Manage User Accounts Using SNMP

Description	Links
Review access requirements for managing user accounts using SNMP	<ul style="list-style-type: none">• “Before You Begin - User Accounts (SNMP)” on page 34
SNMP procedures for configuring user accounts	<ul style="list-style-type: none">• “Configuring User Accounts (SNMP)” on page 34
SNMP procedures for configuring Active Directory settings	<ul style="list-style-type: none">• “Configuring Active Directory Settings” on page 38
SNMP procedure for configuring DNS name server	<ul style="list-style-type: none">• “Manage DNS Name Server Settings (SNMP)” on page 56
SNMP procedure for configuring LDAP settings	<ul style="list-style-type: none">• “Configuring ILOM for LDAP (SNMP)” on page 58
SNMP procedures for configuring LDAP/SSL settings	<ul style="list-style-type: none">• “Configuring ILOM for LDAP/SSL” on page 61
SNMP procedures for configuring RADIUS settings	<ul style="list-style-type: none">• “Configuring RADIUS Settings (SNMP)” on page 71

Related Information

- [“Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts \(CLI\)” on page 8](#)
- [“Managing SNMP Read and Write Access, User Accounts, and SNMP Trap Alerts \(Web\)” on page 19](#)
- *Oracle ILOM 3.0 Daily Management Concepts*, user management
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, manage user accounts
- *Oracle ILOM 3.0 Daily Management Web Procedures*, manage user accounts

Before You Begin - User Accounts (SNMP)

Prior to performing the procedures in this section, you must ensure that the following requirements are met:

- To use SNMP, ensure that all the SNMP properties are correctly set. For more details, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#)
To view user account information, you need the Read Only (o) role enabled.
- To configure user account information, you need the User Management (u) role enabled.
- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user account with read-write (rw) privileges.

Note – For examples of SNMP commands, see [“SNMP Command Examples” on page 263](#).

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will only work as presented if you have Net-SNMP and the Net-SNMP sample applications installed.

Configuring User Accounts (SNMP)

Description	Links
SNMP procedure and valid MIB objects for configuring user accounts	<ul style="list-style-type: none">• “Configure User Accounts (SNMP)” on page 35
SNMP procedure and valid MIB objects for configuring Single Sign On	<ul style="list-style-type: none">• “Configure Single Sign On (SNMP)” on page 37

▼ Configure User Accounts (SNMP)

Note – You can use `get` and `set` commands to configure user account MIB object settings. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the Oracle ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To create a new user account with a user role of Operator, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLocalUserRowStatus.'user1' i 4
ilomCtrlLocalUserRoles.'user1' s "operator"
ilomCtrlLocalUserPassword.'user1' s "password"
```

3. To delete a user account, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLocalUserRowStatus.'user1' i 6
```

Note – For some host OS SNMP clients, you might need to modify the syntax by adding an escape character to the command and changing the quotes to double quotes. For example: `snmpset -v2c -cprivate -mALL gfxsqa-37a ilomCtrlLocalUserRowStatus.\"user1\" i 6`.

The following table describes the User Account SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLocal UserUsername	A local user use rname. It must start with an alphabetical letter and can contain alphabetical letters, digits, hyphens, and underscores, but cannot contain spaces. It cannot be the same as the password.	<i>username</i>	String	None
ilomCtrlLocal UserPassword	A local user password.	<i>password</i>	String	None
ilomCtrlLocal UserRoles	Specifies the role that is associated with a user. The roles can be assigned for the legacy roles of Administrator or Operator, or any of the individual role IDs of a, u, c, r, o and s. The role IDs can be joined together. For example, aucros, where a= admin, u=user, c= console, r=reset, o= read-only, s=service.	administrator, operator, admin(a) , user(u) , console(c) , reset(r) , read-only(o) , service(s)	String	None
ilomCtrlLocal UserRowStatus	This object is used to create a new row or to delete an existing row in the table. This property can be set to either createAndWait(5) or destroy(6), to create and remove a user respectively.	active(1) , notInService(2) , notReady(3) , createAndGo(4) , createAndWait(5) , destroy(6)	Integer	None
ilomCtrlLocal UserCLIMode	An enumerated value that describes the possible CLI modes. The default mode corresponds to the Oracle ILOM DMTF CLP. The alom mode corresponds to the ALOM CMT.	default(1) , alom(2)	Integer	None

▼ Configure Single Sign On (SNMP)

Single Sign On is a convenient authentication service that reduces the number of times you need to enter a password to gain access to Oracle ILOM. Single Sign On is enabled by default. As with any authentication service, authentication credentials are passed over the network. If you do not want this, consider disabling the Single Sign On authentication service.

Note – You can use the `set` command to configure Single Sign On MIB object settings. For a description of the MIB object used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To enable Single Sign On, type:

```
ilomCtrlSingleSignonEnabled.0 i 1
For example:
```

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlSingleSignonEnabled.0 i 1
```

The following table describes the Single Sign On SNMP MIB object.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlSingleSignonEnabled	Specifies whether Single Sign On (SSO) authentication should be enabled on the device. SSO allows tokens to be passed so that it is not necessary to re-enter passwords between different applications. This allows SSO between the system controller (SC) web interface and the service processor (SP) web interface, between the SC command-line interface and the SP command-line interface, and between the SC and SP interfaces and the Java Remote Console application.	true(1), false(2)	Integer	None

Configuring Active Directory Settings

Topic Descriptions	Links
SNMP procedures for configuring Active Directory properties	<ul style="list-style-type: none">• “Manage Active Directory Settings (SNMP)” on page 38• “Manage Active Directory Administrator Groups (SNMP)” on page 43• “Manage Active Directory Operator Group (SNMP)” on page 44• “Manage Active Directory Custom Group (SNMP)” on page 46• “Manage Active Directory User Domains (SNMP)” on page 48• “Manage Active Directory Alternate Server (SNMP)” on page 50• “Manage Server Redundancy (SNMP)” on page 53• “Manage Active Directory DNS Locator (SNMP)” on page 54

▼ Manage Active Directory Settings (SNMP)

Note – You can use the `get` and `set` commands to view and configure Active Directory settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. **Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:**

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. **Refer to the following SNMP command examples:**

- To view the Active Directory state, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryEnabled.0
```

- To enable the Active Directory, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryEnabled.0 i 1
```

- To view the Active Directory port number, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryPortNumber.0
```

- To set the Active Directory port number, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryPortNumber.0 i portnumber
```

- To view the Active Directory default user roles, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryDefaultRoles.0
```

- To set the Active Directory default user roles, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryDefaultRoles.0 s acro
```

- To view the Active Directory certificate file URI, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertFileURI.0
```

- To set the Active Directory certificate file URI, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertFileURI.0 s URI
```

- To view the Active Directory time-out, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryTimeout.0
```

- To set the Active Directory time-out, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryTimeout.0 i 6
```

- To view the Active Directory certificate validation mode, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryStrictCertEnabled.0
```

- To set the Active Directory certificate validation mode, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryStrictCertEnabled.0 i 1
```

- To view the Active Directory certificate file status, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertFileStatus.0
```

- To view the event log setting for the number of messages sent to the event log, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryLogDetail.0
```

- To configure the event log setting so that only the highest priority messages are sent to the event log, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryLogDetail.0 i 2
```

- To view the role that user1 is to have when authenticated through Active Directory, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryDefaultRoles.'user1'
```

- To specify the Admin (a) role for user1 when authenticated via Active Directory, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryDefaultRoles.'user1' s a
```

- To view and clear the certificate information associated with the server when it is set to true, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertClear.0  
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertClear.0 i 0
```

- To view the version of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertVersion.0
```

- To view the serial number of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertserialNo.0
```

- To view the issuer of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertIssuer.0
```

- To view the subject of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertSubject.0
```

- To view the valid start date of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertValidBegin.0
```

- To view the valid end date of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlActiveDirectoryCertValidEnd.0
```

The following table describes the Active Directory Certificates SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActive Directory Enabled	Specifies whether the Active Directory client is enabled.	true(1) , false(2)	Integer	true
ilomCtrlActive DirectoryIP	The IP address of the Active Directory server used as a name service for user accounts.	<i>ipaddress</i>	String	None
ilomCtrlActive Directory PortNumbe	Specifies the port number for the Active Directory client. Specifying 0 as the port means autoselect, while specifying 1 to 65535 configures the actual port.	portnumber Range: 0 to 65535	Integer	None
ilomCtrl Active Directory DefaultRoles	Specifies the role that a user authenticated through Active Directory should have. Setting this property to legacy roles of Administrator or Operator, or any of the individual role IDs of a, u, c, r, o and s will cause the Active Directory client to ignore the schema stored on the Active Directory server. Setting this to none clears the value and indicates that the native Active Directory schema should be used. The role IDs can be joined together. For example, aucros, where a= admin, u=user, c=console, r=reset, o=read-only, and s= service.	administrator , operator , admin(a) , user(u) , console(c) , reset(r) , read-only(o) , service(s) , none	String	None
ilomCtrlActive Directory CertFileURI	This is the URI of a certificate file needed when Strict Certificate Mode is enabled. Setting the URI causes the transfer of the file, making the certificate available immediately for certificate authentication.	<i>URI</i>	String	None

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirectoryTimeout	Specifies the number of seconds to wait before timing out if the Active Directory server is not responding.	Range: 1 to 20 seconds	Integer	4
ilomCtrlActiveDirectoryStrictCertEnabled	Specifies whether the Strict Certificate Mode is enabled for the Active Directory client. If enabled, the Active Directory certificate must be uploaded to the SP so that certificate validation can be performed when communicating with the Active Directory server.	true(1), false(2)	Integer	true
ilomCtrlActiveDirectoryCertFileStatus	A string indicating the status of the certificate file. This is useful in determining whether a certificate file is present or not.	status	String	None

▼ Manage Active Directory Administrator Groups (SNMP)

Note – If you were using the Net-SNMP sample applications, you could use the `snmpget` and `snmpset` commands to configure the Active Directory Administrator Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To view the name of Active Directory administrator group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAdminGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAdminGroupName.2 = STRING:
CN=spAdmins,DC=spc,DC=north,DC=sun,DC=com
```

3. To set the name of Active Directory administrator group ID number 2 to **CN=spAdmins,DC=spc,DC=south,DC=sun,DC=com**, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAdminGroupName.2 s CN=spAdmins,DC=spc,DC=
south,DC=sun,DC=com
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAdminGroupName.2 = STRING:
CN=spAdmins,DC=spc,DC=south,DC=sun,DC=com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAdminGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAdminGroupName.2 = STRING:
CN=spAdmins,DC=spc,DC=south,DC=sun,DC=com
```

The following table describes the Active Directory Administrator Groups SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirAdminGroupId	An integer identifier of the Active Directory Administrator Groups entry.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirAdminGroupName	This string should contain a Distinguished Name that exactly matches one of the group names on the Active Directory server. Any user belonging to one of these groups in this table will be assigned the Oracle ILOM role of Administrator.	<i>name</i> (maximum of 255 characters)	String	None

▼ Manage Active Directory Operator Group (SNMP)

Note – You can use the get and set commands to configure the Active Directory Operator Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To view the name of Active Directory operator group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirOperatorGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirOperatorGroupName.2 =
STRING: ad-oper-group-ent-2
```

3. To set the name of Active Directory operator group ID number 2 to new-name-2, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirOperatorGroupName.2 s new-name-2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirOperatorGroupName.2 =
STRING: new-name-2
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirOperatorGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirOperatorGroupName.2 =
STRING: new-name-2
```

The following table describes the Active Directory Operator Group SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirOperatorGroupId	An integer identifier of the Active Directory Operator Groups entry.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirOperatorGroupName	This string should contain a Distinguished Name that exactly matches one of the group names on the Active Directory server. Any user belonging to one of these groups in this table will be assigned the Oracle ILOM role of Operator.	name (maximum of 255 characters)	String	None

▼ Manage Active Directory Custom Group (SNMP)

Note – You can use the `get` and `set` commands to configure the Active Directory Custom Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. To view the name of Active Directory custom group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupName.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupName.2 =  
STRING: CN=SpSuperCust,OU=Groups,DC=johns,DC=sun,DC=com
```

3. To set the name of Active Directory custom group ID number 2 to **CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com**, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupName.2 s CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupName.2 =  
STRING: CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com  
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupName.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupName.2 =  
STRING: CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com
```

4. To view the roles of Active Directory custom group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupRoles.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupRoles.2 =  
STRING: "aucro"
```

5. To set the roles of Active Directory custom group ID number 2 to User Management and Read Only (u,o), type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupRoles.2 s "uo"  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupRoles.2 =  
STRING: "uo"  
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirCustomGroupRole.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirCustomGroupRoles.2 =  
STRING: "uo"
```

The following table describes the Active Directory Custom Group SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirCustomGroupId	An integer identifier of the Active Directory Custom Groups entry.	1 to 5 This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirCustomGroupName	This string should contain a Distinguished Name that exactly matches one of the group names on the Active Directory server. Any user belonging to one of these groups in this table will be assigned the Oracle ILOM role based on the entry's configuration for roles.	<i>name</i> (maximum of 255 characters)	String	None
ilomCtrlActiveDirCustomGroupRoles	Specifies the role that a user authenticated via Active Directory should have. Setting this property to legacy roles of Administrator or Operator, or any of the individual role IDs of a, u, c, r, o and s will cause the Active Directory client to ignore the schema stored on the Active Directory server. Setting this object to none clears the value and indicates that the native Active Directory schema should be used. The role IDs can be joined together. For example, aucros, where a= admin, u=user, c=console, r= reset, o=read-only, and s= service.	administrator , operator , admin(a) , user(u) , console(c) , reset(r) , read-only(o) , service(s) , none	String	None

▼ Manage Active Directory User Domains (SNMP)

Note – You can use the `get` and `set` commands to configure the Active Directory User Domain settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To view the name of Active Directory user domain ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirUserDomain.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirUserDomain.2 = STRING:
<USERNAME>@davidc.example.oracle.com
```

3. To set the name of Active Directory user domain ID number 2 to <USERNAME>@johns.example.oracle.com, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirUserDomain.2 s
"<USERNAME>@johns.example.oracle.com"
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirUserDomain.2 = STRING:
<USERNAME>@johns.example.oracle.com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirUserDomain.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirUserDomain.2 = STRING:
<USERNAME>@johns.example.oracle.com
```

The following table describes the Active Directory User Domains SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirUserDomainId	An integer identifier of the Active Directory domain.	1 to 5 This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirUserDomain	This string should exactly match with an authentication domain on the Active Directory server. This string should contain a substitution string (<USERNAME>), which will be replaced with the user's login name during authentication. Either the principle or Distinguished Name format is allowed.	name (maximum of 255 characters)	String	None

▼ Manage Active Directory Alternate Server (SNMP)

Note – You can use the `get` and `set` commands to set the values of MIB object properties to configure the Active Directory Alternate Server settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. Refer to the following SNMP command examples:

- To view the IP address of Active Directory alternate server ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirAlternateServerIp.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerIp.2 =  
IpAddress: 10.7.143.236
```

- To set the IP address of Active Directory alternate server ID number 2 to 10.7.143.246, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirAlternateServerIp.2 a 10.7.143.246  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerIp.2 =  
IpAddress: 10.7.143.246  
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirAlternateServerIp.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerIp.2 =  
IpAddress: 10.7.143.246
```

- To view the port number of Active Directory alternate server ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirAlternateServerPort.2  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerPort.2 =  
INTEGER: 636
```


- To set the port number of Active Directory alternate server ID number 2 to 639, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerPort.2 i 639
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerPort.2 =
INTEGER: 639
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerIp.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerPort.2 =
INTEGER: 639
```

- To view the certificate status of Active Directory alternate server ID number 2, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertStatus.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerCertStatus.
2 = STRING: certificate not present
```

- To view the certificate URI of Active Directory alternate server ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertURI.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirAlternateServerCertURI.2 =
STRING: none
```

- To clear the certificate information associated with the server when it is set to true, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertClear.0 i 1
```

- To view the certificate version of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertVersion.0
```

- To view the serial number of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertSerialNo.0
```

- To view the issuer of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertIssuer.0
```

- To view the subject of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertSubject.0
```

- To view the valid start date of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertValidBegin.0
```

- To view the valid end date of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlActiveDirAlternateServerCertValidEnd.0
```

The following table describes the Active Directory Alternate Server SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirAlternateServerId	An integer identifier of the Active Directory alternate server table.	1 to 5 This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirAlternateServerIP	The IP address of the Active Directory alternate server used as a name service for user accounts.	<i>ipaddress</i>	String	None

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirAlternateServerPort	Specifies the port number for the Active Directory alternate server. Specifying 0 as the port indicates that autoselect will use the well known port number. Specifying 1-65535 explicitly sets the port number.	<i>portnumber</i> (range: 0 to 65535)	Integer	None
ilomCtrlActiveDirAlternateServerCertStatus	A string indicating the status of the certificate file. This is useful in determining whether a certificate file is present or not.	<i>status</i> (maximum size: 255 characters)	String	None
ilomCtrlActiveDirAlternateServerCertURI	This is the URI of a certificate file needed when Strict Certificate Mode is enabled. Setting the URI causes the transfer of the file, making the certificate available immediately for certificate authentication. Additionally, either <i>remove</i> or <i>restore</i> are supported for direct certificate manipulation.	<i>URI</i>	String	None

▼ Manage Server Redundancy (SNMP)

Note – You can use the *get* and *set* commands to view and configure redundancy settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the status of the server in a redundant configuration, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRedundancyStatus.0
```

- To view the property that controls whether the server is to be promoted or demoted from active or standby status, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRedundancyAction.0
```

- To promote a redundant server from standby to active status, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRedundancyAction.0 i 2
```

- To view the FRU name of the chassis monitoring module (CMM) on which this agent is running, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRedundancyFRUName.0
```

▼ Manage Active Directory DNS Locator (SNMP)

Note – You can use the get and set commands to configure the Active Directory DNS Locator settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress  
Password: password
```

2. To view the state of Active Directory DNS locator, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlActiveDirDnsLocatorEnabled.0  
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorEnabled.0 =  
INTEGER: false(2)
```

3. To set the state of Active Directory DNS locator ID number 2 to enabled, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirDnsLocatorEnabled.0 i 1
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorEnabled.0 =
INTEGER: true(1)
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirDnsLocatorEnabled.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorEnabled.2 =
INTEGER: true(1)
```

4. To view the service name of Active Directory DNS locator ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirDnsLocatorQueryService.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorQueryService.2 =
STRING: _ldap._tcp.dc._msdcs.<DOMAIN>.<PORT:636>
```

5. To set the service name and port number of Active Directory DNS locator ID number 2, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirDnsLocatorQueryService.2 s
"_ldap._tcp.pdc._msdcs.<DOMAIN>.<PORT:936>"
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorQueryService.2 =
STRING: _ldap._tcp.pdc._msdcs.<DOMAIN>.<PORT:936>
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlActiveDirDnsLocatorQueryService.2
SUN-ILOM-CONTROL-MIB::ilomCtrlActiveDirDnsLocatorQueryService.2 =
STRING: _ldap._tcp.pdc._msdcs.<DOMAIN>.<PORT:936>
```

The following table describes the Active Directory DNS Locator SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlActiveDirDnsLocatorEnabled	Specifies whether or not the Active Directory DNS locator functionality is enabled.	true(1), false(2)	Integer	false
ilomCtrlActiveDirDnsLocatorQueryId	An integer identifier of the Active Directory DNS Locator Query entry.	1 to 5 This object is not accessible for reading or writing.	Integer	None
ilomCtrlActiveDirDnsLocatorQueryService	The service name that is used to perform the DNS query. The name can contain <DOMAIN> as a substitution marker, being replaced by the domain information associated for the user at the time of authentication. The service name can also contain <PORT:>, which can be used to override any learned port information, if necessary. For example, <PORT:636> can be specified for the standard LDAP/SSL port 636.	name (maximum of 255 characters)	String	None

▼ Manage DNS Name Server Settings (SNMP)

Note – You can use the `get` and `set` commands to view and configure DNS name server settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view and specify the name server for DNS, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSNameServers.0  
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSNameServers.0 s 'nameservername'
```

- To view and specify the search path for DNS, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSSearchPath.0  
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSSearchPath.0 s 'searchpath'
```

- To view state of DHCP autodns for DNS, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSdhcpAutoDns.0
```

- To set the state of DHCP autodns for DNS to enabled, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSdhcpAutoDns.0 i 1
```

- To view the number of seconds to wait before timing out if the server does not respond, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSTimeout.0
```

- To set the number of seconds to wait before timing out if the server does not respond to 5, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSTimeout.0 i 5
```

- To view the number of times a request is attempted again after a time-out, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSRetries.0
```

- To set the number of times a request is attempted again after a time-out to 5, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlDNSRetries.0 i 5
```

Configuring ILOM for LDAP (SNMP)

Topic Descriptions	Links
SNMP procedure for configuring ILOM LDAP properties	<ul style="list-style-type: none">• “Configure LDAP Settings (SNMP)” on page 58

▼ Configure LDAP Settings (SNMP)

Note – You can use the `get` and `set` commands to configure ILOM for LDAP. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view whether the LDAP server is enabled to authenticate LDAP users, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapEnabled.0
```

- To set the LDAP server state to enabled to authenticate LDAP users, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapEnabled.0 i 1
```

- To view the LDAP server IP address, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapServerIP.0
```

- To set the LDAP server IP address, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapServerIP.0 a ipaddress
```


- To view the LDAP server port number, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapPortNumber.0
```

- To set the LDAP server port number, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapPortNumber.0 i 389
```

- To view the LDAP server Distinguished Name, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapBindDn.0
```

- To set the LDAP server Distinguished Name, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapBindDn.0 s ou=people,ou=sales,dc=sun,dc=com
```

- To view the LDAP server password, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapBindPassword.0
```

- To set the LDAP server password, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapBindPassword.0 s password
```

- To view the branch of your LDAP server on which user searches are made, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSearchBase.0
```

- To set the branch of your LDAP server on which to search for users, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSearchBase.0 s ldap_server_branch
```

- To view the LDAP server default role, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapDefaultRoles.0
```

- To set the LDAP server default role to Administrator, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapDefaultRoles.0 s administrator
```

The following table describes the LDAP Settings SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapEnabled	Specifies whether the LDAP client is enabled.	true(1) , false(2)	Integer	false
ilomCtrlLdapServerIP	The IP address of the LDAP server used as a name service for user accounts.	<i>ipaddress</i>	String	None
ilomCtrlLdapPortNumber	Specifies the port number for the LDAP client.	Range: 0..65535	Integer	389
ilomCtrlLdapBindDn	The Distinguished Name (DN) for the read-only proxy user used to bind to the LDAP server. For example: "cn=proxyuser,ou=people,dc=sun,dc=com"	<i>distinguished_name</i>	String	None
ilomCtrlLdapBindPassword	The password of a read-only proxy user that is used to bind to the LDAP server. This property is essentially write-only. The write-only access level is no longer supported as of SNMP v2. This property must return a null value when read.	<i>password</i>	String	None
ilomCtrlLdapSearchBase	A search base in the LDAP database below which to find users. For example: "ou=people,dc=sun,dc=com"	The branch of your LDAP server on which to search for users	String	None
ilomCtrlLdapDefaultRoles	Specifies the role that a user authenticated via LDAP should have. This property supports the legacy roles of Administrator or Operator, or any of the individual role ID combinations of a, u, c, r, o and s. For example, aucros, where a=admin, u=user, c=console, r=reset, o=read-only, and s=service.	administrator, operator, admin(a) , user(u) , console(c) , reset(r) , read-only(o) , service(s)	String	None

Configuring ILOM for LDAP/SSL

Topic Descriptions	Links
SNMP procedures for configuring LDAP/SSL settings	<ul style="list-style-type: none">• “Manage LDAP/SSL Certificate (SNMP)” on page 61• “Manage LDAP/SSL Administrator Group (SNMP)” on page 62• “Manage LDAP/SSL Operator Group (SNMP)” on page 63• “Manage LDAP/SSL Custom Group (SNMP)” on page 65• “Manage LDAP/SSL User Domain (SNMP)” on page 67• “Manage LDAP/SSL Alternate Server (SNMP)” on page 68

▼ Manage LDAP/SSL Certificate (SNMP)

Note – You can use the `get` and `set` commands to view and configure LDAP/SSL certificate settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. Refer to the following SNMP command examples:

- To clear the certificate information associated with the server when it is set to true, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileClear.0 i 0
```

- To view the certificate version of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileVersion.0
```

- To view the serial number of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileSerialNo.0
```

- To view the issuer of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileIssuer.0
```

- To view the subject of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileSubject.0
```

- To view the valid start date of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileValidBegin.0
```

- To view the valid end date of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslCertFileValidEnd.0
```

▼ Manage LDAP/SSL Administrator Group (SNMP)

Note – You can use the get and set commands to configure the LDAP/SSL Administrator Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. Refer to the following SNMP command examples:

- To view the name of LDAP/SSL administrator group ID number 3, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslAdminGroupName.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAdminGroupName.3 = STRING:
CN=SpSuperAdmin,OU=Groups,DC=davidc,DC=example,DC=sun,DC=com
```

- To set the name of LDAP/SSL administrator group ID number 3 to CN=SpSuperAdmin,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslAdminGroupName.3 s CN=SpSuperAdmin,OU=Groups,DC=
tomp,DC=example,DC=sun,DC=com
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAdminGroupName.3 = STRING:
CN=SpSuperAdmin,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslAdminGroupName.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAdminGroupName.3 = STRING:
CN=SpSuperAdmin,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com
```

The following table describes the LDAP/SSL Administrator Group SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapSslAdminGroup Id	An integer identifier of the LDAP/SSL AdminGroup entry.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlLdapSslAdminGroup Name	This string should contain a Distinguished Name that exactly matches one of the group names on the LDAP/SSL server. Any user belonging to one of these groups in this table will be assigned the ILOM role of Administrator.	<i>name</i> (maximum of 255 characters)	String	None

▼ Manage LDAP/SSL Operator Group (SNMP)

Note – You can use the `get` and `set` commands to configure the LDAP/SSL Operator Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the name of LDAP/SSL operator group ID number 3, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslOperatorGroupName.3SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslOperatorGroupName.3 = STRING: CN=SpSuperOper,OU=Groups,DC=davidc,DC=example,DC=sun,DC=com
```

- To set the name of Active Directory operator group ID number 3 to CN=SpSuperAdmin, OU=Groups, DC=tomp, DC=example, DC=sun, DC=com, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslOperatorGroupName.3 s CN=SpSuperOper,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslOperatorGroupName.3 =
STRING: CN=SpSuperOper,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslOperatorGroupName.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslOperatorGroupName.3 =
STRING: CN=SpSuperOper,OU=Groups,DC=tomp,DC=example,DC=sun,DC=com
```

The following table describes the LDAP/SSL Operator Group SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapSslOperatorGroupId	An integer identifier of the LDAP/SSL Operator Group entry.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlLdapSslOperatorGroupName	This string should contain a Distinguished Name that exactly matches one of the group names on the LDAP/SSL server. Any user belonging to one of these groups in this table will be assigned the ILOM role of Operator.	name (maximum of 255 characters)	String	None

▼ Manage LDAP/SSL Custom Group (SNMP)

Note – You can use the `get` and `set` commands to configure the LDAP/SSL Custom Groups settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the name of LDAP/SSL custom group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupName.2 = STRING:
CN=SpSuperCust,OU=Groups,DC=johns,DC=sun,DC=com
```

- To set the name of LDAP/SSL custom group ID number 2 to CN=SpSuperCust, OU=Groups, DC=bills, DC=sun, DC=com, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupName.2 s CN=SpSuperCust,OU=Groups,DC=
bills,DC=sun,DC=com
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupName.2 = STRING:
CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupName.2
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupName.2 = STRING:
CN=SpSuperCust,OU=Groups,DC=bills,DC=sun,DC=com
```

- To view the roles of LDAP/SSL custom group ID number 2, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupRoles.2
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupRoles.2 = STRING:
"aucro"
```

- To set the roles of LDAP/SSL custom group ID number 2 to User Management and Read Only (u,o), type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupRoles.2 s "uo"
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupRoles.2 = STRING:
"uo"
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslCustomGroupRoles.2
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslCustomGroupRoles.2 = STRING:
"uo"
```

The following table describes the LDAP/SSL Custom Group SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapSslCustomGroupId	An integer identifier of the LDAP/SSL custom group entry.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlLdapSslCustomGroupName	This string should contain a Distinguished Name that exactly matches one of the group names on the LDAP/SSL server. Any user belonging to one of these groups in this table will be assigned the ILOM role based on the entry's configuration for roles.	<i>name</i> (maximum of 255 characters)	String	None
ilomCtrlLdapSslCustomGroupRoles	Specifies the role that a user authenticated through LDAP/SSL should have. Setting this property to legacy roles of Administrator or Operator, or any of the individual role IDs of a, u, c, r, o and s will cause the LDAP/SSL client to ignore the schema stored on the LDAP/SSL server. Setting this object to none clears the value and indicates that the native LDAP/SSL schema should be used. The role IDs can be joined together. For example, aucros, where a=admin, u=user, c=console, r=reset, o= read-only, and s=service.	administrator, operator, admin(a), user(u), console(c), reset(r), read-only(o), service(s), none	String	None

▼ Manage LDAP/SSL User Domain (SNMP)

Note – You can use the `get` and `set` commands to configure the LDAP/SSL User Domain settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the name of LDAP/SSL user domain ID number 3, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslUserDomain.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslUserDomain.3 = STRING: CN=
<USERNAME>, CN=Users, DC=davidc, DC=example, DC=sun, DC=com
```

- To set the name of LDAP/SSL user domain ID number 3 to CN=<USERNAME>, CN=Users, DC=tomp, DC=example, DC=sun, DC=com, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslUserDomain.3 s CN=<USERNAME>, CN=Users, DC=tomp, DC=
example, DC=sun, DC=com
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslUserDomain.3 = STRING: CN=
<USERNAME>, CN=Users, DC=tomp, DC=example, DC=sun, DC=com
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslUserDomain.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslUserDomain.3 = STRING: CN=
<USERNAME>, CN=Users, DC=tomp, DC=example, DC=sun, DC=com
```

The following table describes the LDAP/SSL User Domain SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapSslUserDomainId	An integer identifier of the LDAP/SSL domain.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlLdapSslUserDomain	This string should exactly match with an authentication domain on the LDAP/SSL server. This string should contain a substitution string (<USERNAME>), which will be replaced with the user's login name during authentication. Either the principle or Distinguished Name format is allowed.	<i>name</i> (maximum of 255 characters)	String	None

▼ Manage LDAP/SSL Alternate Server (SNMP)

Note – You can use the `get` and `set` commands to configure the LDAP/SSL Alternate Server settings. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

Password: *password*

2. Refer to the following SNMP command examples:

- To view the IP address of LDAP/SSL alternate server ID number 3, type:

```
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress
ilomCtrlLdapSslAlternateServerIp.3
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAlternateServerIp.3 =
IpAddress: 10.7.143.236
```

- To set the IP address of LDAP/SSL alternate server ID number 3 to 10.7.143.246, type:

```
% snmpset -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerIp.3 a 10.7.143.246  
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAlternateServerIp.3 =  
IpAddress: 10.7.143.246  
% snmpget -v1 -cprivate -mALL SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerIp.3  
SUN-ILOM-CONTROL-MIB::ilomCtrlLdapSslAlternateServerIp.3 =  
IpAddress: 10.7.143.246
```

- To view and clear the certificate information associated with the alternate server when it is set to true, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertClear.0  
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertClear.0 i 0
```

- To view the alternate server certificate version of the certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertVersion.0
```

- To view the serial number of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertSerialNo.0
```

- To view the issuer of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertIssuer.0
```

- To view the subject of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertSubject.0
```

- To view the valid start date of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlLdapSslAlternateServerCertValidBegin.0
```

- To view the valid end date of the alternate server certificate file, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlLdapSslAlternateServerCertValidEnd.0
```

The following table describes the LDAP/SSL Alternate Server SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlLdapSslAlternateServerId	An integer identifier of the LDAP/SSL alternate server table.	1 to 5 Note - This object is not accessible for reading or writing.	Integer	None
ilomCtrlLdapSslAlternateServerIP	The IP address of the LDAP/SSL alternate server used as directory server for user accounts.	<i>ipaddress</i>	String	None
ilomCtrlLdapSslAlternateServerPort	Specifies the port number for the LDAP/SSL alternate server. Specifying 0 as the port indicates that auto-select will use the well-known port number. Specifying 1-65535 explicitly sets the port number.	<i>portnumber</i> (range: 0 to 65535)	Integer	None
ilomCtrlLdapSslAlternateServerCertStatus	A string indicating the status of the certificate file. This is useful in determining whether a certificate file is present or not.	<i>status</i> (maximum size: 255 characters)	String	None
ilomCtrlLdapSslAlternateServerCertURI	This is the URI of a certificate file needed when Strict Certificate Mode is enabled. Setting the URI causes the transfer of the file, making the certificate available immediately for certificate authentication. Additionally, either <i>remove</i> or <i>restore</i> are supported for direct certificate manipulation.	<i>URI</i>	String	None

Configuring RADIUS Settings (SNMP)

Topic Descriptions	Links
SNMP procedure for configuring ILOM RADIUS properties	<ul style="list-style-type: none">• “Configure RADIUS Settings (SNM)” on page 71

▼ Configure RADIUS Settings (SNM))

Note – Before completing this procedure, collect the appropriate information about your RADIUS environment. You can use the `get` and `set` commands to configure RADIUS. For a description of the MIB objects used in this procedure, see the table that follows the procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view whether the RADIUS server is enabled to authenticate RADIUS users, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlRadiusEnabled.0
```

- To set the RADIUS server state to enabled to authenticate RADIUS users, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlRadiusEnabled.0 i 1
```

- To view the RADIUS server IP address, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlRadiusServerIP.0
```

- To set the RADIUS server IP address, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusServerIP.0 a ipaddress
```

- To view the RADIUS server port number, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusPortNumber.0
```

- To set the RADIUS server port number, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusPortNumber.0 i portnumber
```

- To view the RADIUS server shared secret, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusSecret.0
```

- To set the RADIUS server shared secret, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusSecret.0 s secret
```

- To view the RADIUS server default user roles, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusDefaultRoles.0
```

- To set the RADIUS server default user roles to console, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlRadiusDefaultRoles.0 s c
```

The following table describes the RADIUS SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlRadiusEnabled	Specifies whether or not the RADIUS client is enabled.	true(1) , false(2)	Integer	false
ilomCtrlRadiusServerIP	The IP address of the RADIUS server used as a name service for user accounts.	ipaddress	String	None
ilomCtrlRadiusPortNumber	Specifies the port number for the RADIUS client.	portnumber (range: 0 to 65535)	Integer	1812
ilomCtrlRadiusSecret	The shared secret encryption key that is used to encrypt traffic between the RADIUS client and server.	secret (maximum length: 255 characters)	String	None
ilomCtrlRadiusDefaultRoles	Specifies the role that a user authenticated through RADIUS should have. This property supports the legacy roles of Administrator or Operator, or any of the individual role ID combinations of a, u, c, r, o and s. For example, aucros, where a=admin, u=user, c=console, r=reset, o=read-only, and s=service.	administrator, operator, admin(a), user(u), console(c), reset(r), read-only(o), service(s)	String	None

Manage Component Information and Email Alerts (SNMP)

Description	Links
Review ILOM requirements for managing component information and email alerts using SNMP	<ul style="list-style-type: none">• “Before You Begin - Component Information (SNMP)” on page 76
SNMP view component procedures	<ul style="list-style-type: none">• “Viewing Component Information” on page 76
SNMP configuration procedure for managing clock settings, syslog and alert rules	<ul style="list-style-type: none">• “Managing Clock Settings, Event Log, Syslog Receiver, and Alert Rules” on page 78
SNMP configuration procedure for SMTP client for Email notification alerts	<ul style="list-style-type: none">• “Configuring SMTP Client for Email Alert Notifications” on page 84
SNMP configuration procedure for alerts	<ul style="list-style-type: none">• “Configuring Email Alert Settings (SNMP)” on page 86
SNMP configuration procedure for Telemetry Harness Daemon	<ul style="list-style-type: none">• “Configuring Telemetry Harness Daemon (SNMP)” on page 87

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, system monitoring and alert management
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, managing alerts
- *Oracle ILOM 3.0 Daily Management Web Procedures*, managing alerts

Before You Begin - Component Information (SNMP)

- Before you can use SNMP to view and configure ILOM settings, you must configure SNMP. For more information, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#).
- When executing the `snmpset` command, you need to use a v1/v2c community or a v3 user with read/write (rw) privileges.

Note – For examples of SNMP commands, see [“SNMP Command Examples” on page 263](#).

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work only as presented if you have Net-SNMP and the Net-SNMP sample applications installed.

Viewing Component Information

Topic Descriptions	Links
SNMP procedure for viewng ILOM component information	<ul style="list-style-type: none">• “View Component Information” on page 76

▼ View Component Information

Note – You can use `get` commands to view component information. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ip_address
Password: password
```

2. To view the firmware revision, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
entPhysicalFirmwareRev.1
```

The following table describes the Component Information SNMP MIB objects.

MIB Object	Description	Values	Type	Default
entPhysicalName	The textual name of the physical entity.	Size: 0..255	String	Zero-length string
entPhysicalDescr	A textual description of the physical entity.	Size: 0..255	String	None
entPhysicalContainedIn	The value of entPhysicalIndex for the physical entity that <i>contains</i> this physical entity. A value of 0 indicates this physical entity is not contained in any other physical entity.	Range: 0..2147483647	Integer	None
entPhysicalClass	An indication of the general hardware type of the physical entity.	other(1), unknown(2), chassis(3), backplane(4), container(5), powerSupply(6), fan(7), sensor(8), module(9), port(10), stack(11)	Integer	None
entPhysicalFirmwareRev	The vendor-specific firmware revision string for the physical entity.	Size: 0..255	String	Zero-length string

Managing Clock Settings, Event Log, Syslog Receiver, and Alert Rules

Description	Links
SNMP procedure and valid MIB objects to view and set clock settings	<ul style="list-style-type: none">• “View and Set Clock Settings” on page 78
SNMP procedure and valid MIB objects to view and clear the ILOM event log	<ul style="list-style-type: none">• “View and Clear the ILOM Event Log” on page 79
SNMP procedure and valid MIB objects to configure remote syslog receiver IP addresses	<ul style="list-style-type: none">• “Configure Remote Syslog IP Destinations” on page 81
SNMP procedure and valid MIB objects to configure alert rules	<ul style="list-style-type: none">• “Configure Severity Level Alert Rule” on page 82

▼ View and Set Clock Settings

Note – You can use the `get` and `set` commands to view and set clock settings with respect to Network Time Protocol (NTP) synchronization. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ip_address
Password: password
```

2. Refer to the following SNMP commands for examples:

- To view the NTP server state, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlNTPEnabled.0
```

- To set the NTP server state to enabled, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlNTPEnabled.0 i 1
```

- To view the date and time of the device, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlDateAndTime.0
```

- To set the date and time of the device, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlDateAndTime.0 s 2008-3-24,4:59:47.0
```

The following table describes the valid SNMP MIB objects for Oracle ILOM clock properties.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlDateAndTime	The date and time of the device.	<i>date/time</i>	String	None
ilomCtrlNTPEnabled	Specifies whether the Network Time Protocol is enabled.	true(1) , false(2)	Integer	false
ilomCtrlTimezone	The configured time zone string.	Size: 0..255	String	None

▼ View and Clear the ILOM Event Log

Note – You can use the `get` command to view the ILOM event log and the `set` command to configure the ILOM event log. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ip_address
Password: password
```

2. To view the ILOM event log type for an event log with a record ID of 2, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlEventLogType.2
```

3. To clear the ILOM event log, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlEventLogClear.0 i 1
```

The following table describes the ILOM Event Logs SNMP MIB objects.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlEventLogRecordID	The record number for a given event log entry. Note - This object is not accessible.	Range: 1..10000	Integer	None
ilomCtrlEventLogType	An integer representing the type of event. Note - This object is read-only.	log(1), action2), fault(3), state(4), repair(5)	Integer	None
ilomCtrlEventLogTimestamp	The date and time that the event log entry was recorded. Note - This object is read-only.	date/time	String	None
ilomCtrlEventLogClass	An integer representing the class of event. Note - This object is read-only.	audit(1), ipmi(2), chassis(3), fma(4), system(5), pcm(6)	Integer	None
ilomCtrlEventLogSeverity	The event severity corresponding to the given log entry. Note - This object is read-only.	disable(1), critical(2), major(3), minor(4), down(5)	Integer	None

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlEventLogDescription	A textual description of the event. Note - This object is read-only.	<i>description</i>	String	None
ilomCtrlEventLogClear	Setting this object to true clears the event log.	true(1), false(2)	Integer	None

▼ Configure Remote Syslog IP Destinations

Note – You can use the `get` and `set` commands to view and set IP addresses for a remote syslog receiver. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ip_address
Password: password
```

2. To view a remote syslog destination IP address, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlRemoteSyslogDest1.0
```

3. To set a remote syslog destination IP address, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlRemoteSyslogDest1.0 s ip_address
```

The following table describes the Syslog IP Destinations SNMP MIB objects.

MIB Object	Description	Values	Type	Default
ilomCtrlRemoteSyslogDest1	The IP address of the first remote syslog destination (log host).	<i>ip_address</i>	String	None
ilomCtrlRemoteSyslogDest2	The IP address of the second remote syslog destination (log host).	<i>ip_address</i>	String	None

▼ Configure Severity Level Alert Rule

Note – You can use the `get` and `set` commands to view and configure alert rule configurations. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ip_address
```

```
Password: password
```

2. To view the severity level for the alert rule with an alert ID of 2, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlAlertSeverity.2
```

3. To set the severity level to critical for the alert rule with an alert ID of 2, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address
ilomCtrlAlertSeverity.2 i 2
```

The following table describes the Alert Rule Severity Level SNMP MIB objects.

Note – Oracle ILOM does not support alert level filtering for SNMP traps. To enable the sending of an SNMP trap (but not filter the SNMP trap by alert level) you can choose one of the following severity levels: Minor, Major, Critical, or Down. To disable the sending of an SNMP trap, you must choose the Disabled option.

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlAlert ID	An integer ID associated with a given alert rule. Note - This object is not accessible.	Range: 0..65535	Integer	None
ilomCtrlAlert Severity	Specifies the minimum event severity that should trigger an alert for a given class.	disable(1), critical(2), major(3), minor(4), down(5)	Integer	None

MIB Object	Description	Allowed Values	Type	Default
ilomCtrlAlertType	Specifies the type of notification for a given alert. If the type is <code>snmptrap(2)</code> or <code>ipmipet(3)</code> , the <code>ilomCtrlAlertDestinationip</code> must be specified. If the type is <code>email(1)</code> , the <code>ilomCtrlAlertDestinationEmail</code> must be specified.	<code>email(1)</code> <code>snmptrap(2)</code> <code>ipmipet(3)</code> <code>remotesyslog(4)</code>	Integer	None
ilomCtrlAlertDestinationip	Specifies the IP address to send alert notifications to when the alert type is <code>snmptrap(2)</code> , <code>ipmipet(3)</code> , or <code>remotesyslog(4)</code> .	<i>ip_address</i>	String	None
ilomCtrlAlertDestinationEmail	Specifies the email address to send alert notifications to when the alert type is <code>email(1)</code> .	<i>email address</i> , size: 0..255	String	None
ilomCtrlAlertSNMPVersion	Specifies the version of SNMP trap that should be used for the given alert rule.	<code>v1(1)</code> , <code>v2c(2)</code> , <code>v3(3)</code>	Integer	None
ilomCtrlAlertSNMPCommunityOrUsername	Specifies the community string to be used when the <code>ilomCtrlAlertSNMPVersion</code> property is set to <code>v1(1)</code> or <code>v2c(2)</code> . Specifies the SNMP user name to use when the <code>ilomCtrlAlertSNMPVersion</code> is set to <code>v3(3)</code> .	Size: 0..255	String	None
ilomCtrlAlertEmailEventClassFilter	A class name or <code>all</code> to filter emailed alerts on.	Size: 0..255	String	None
ilomCtrlAlertEmailEventTypeFilter	A class name or <code>all</code> to filter emailed alerts on.	Size 0..255	String	None

Configuring SMTP Client for Email Alert Notifications

Description	Links
Procedure for configuring SMTP email alert notification	<ul style="list-style-type: none">• “Configure SMTP Client for Alert Notification (SNMP)” on page 84

▼ Configure SMTP Client for Alert Notification (SNMP)

Before You Begin

- To generate configured email notification alerts, you must enable the ILOM client to act as an SMTP client to send the email alert messages. To enable the ILOM client as an SMTP client, you must specify the IP address and port number of an outgoing SMTP email server that will process the email notifications.
- Prior to enabling the ILOM client as an SMTP client, gather the IP address and port number of the outgoing SMTP email server.
- You can use the `get` and `set` commands to configure the SMTP client. For a description of the MIB objects used in this procedure, see Valid SMTP Client MIB Objects and the SUN-ILOM-CONTROL-MIB.

Note – For a description of valid MIB objects for this procedure, see the table following this procedure.

To configure SMTP Client properties in Oracle ILOM, follow these steps:

1. **Log in to a host that has an SNMP tool and the Oracle ILOM MIBs installed. For example, type:**

```
ssh username@snmp_manager_ip_address  
Password: password
```

2. **Refer to the following SNMP commands for examples:**

- To view a SMTP client state, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPEnabled.0
```

- To set a SMTP client state to enabled, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPEnabled.0 i 1
```

- To view a SMTP server IP address, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPServerip.0
```

- To set a SMTP server IP address, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPServerip.0 s ip_address
```

- To view a SMTP client port number, type:

```
% snmpget -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPPortNumber.0
```

- To set a SMTP client port number, type:

```
% snmpset -v2c -cprivate -mALL SNMP_agent_ip_address  
ilomCtrlSMTPPortNumber.0 i 25
```

- To view an optional format to identify the sender or the “from” address, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSMTPCustomSender.0
```

- To configure an optional format to identify the sender or the “from” address, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSMTPCustomSender.0 s 'ilom-alert@HOSTNAME.abc.com'
```

The following table describes the SMTP Email Alert Notification SNMP MIB objects.

MIB Object	Property	Allowed Values	Type	Default
ilomCtrlSMTP Enabled	Specifies whether or not the SMTP client is enabled.	true(1) , false(2)	Integer	false
ilomCtrlSMTP Serverip	The IP address of the SMTP server used as a name service for user accounts.	<i>ip_address</i>	String	None
ilomCtrlSMTP PortNumber	Specifies the port number for the SMTP client.	Range: 0..65535	Integer	None

Configuring Email Alert Settings (SNMP)

Description	Links
SNMP procedure to view or configure email alert settings in ILOM	<ul style="list-style-type: none"> • “Manage Email Alert Settings (SNMP)” on page 86

▼ Manage Email Alert Settings (SNMP)

Note – You can use the `get` and `set` commands to view and configure email alert settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the optional format used to identify the sender or the “from” address, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlAlertEmailCustomSender.0
```

- To set the optional format used to identify the sender or the “from” address, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlAlertEmailCustomSender.0 s
'ilom-alert@HOSTNAME.abc.com'
```

- To view an optional string that can be added to the beginning of the message body, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlAlertEmailMessagePrefix.0
```

- To define an optional string (for example: BeginMessage) that can be added to the beginning of the message body, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlAlertEmailMessagePrefix.0 s 'BeginMessage'
```



Configuring Telemetry Harness Daemon (SNMP)

Description	Links
SNMP procedure for configuring telemetry harness daemon settings	<ul style="list-style-type: none"> • “Manage Telemetry Harness Daemon Settings (SNMP)” on page 88

▼ Manage Telemetry Harness Daemon Settings (SNMP)

Note – You can use the `get` and `set` commands to view and configure Telemetry Harness Daemon (THD) settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. Refer to the following SNMP command examples:

- To view the state of the THD daemon, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdState.0
```

- To view the control action for THD daemon, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdAction.0
```

- To set the control action for THD daemon to suspend, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdAction.0 i 1
```

- To view the description of the THD module named THDMod1, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdModuleDesc.'THDMod1'
```

- To view the state of the THD module named THDMod1, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdModuleState.'THDMod1'
```

- To view the control action for the THD module named THDMod1, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdModuleAction.'THDMod1'
```

- To set the control action for the THD module named THDMod1 to suspend, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdModuleAction.0 i 1
```

- To view the state of the THD instance named myTHDinstance that is in the THD class named myTHDclass, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdInstanceState.'myTHDclass.myTHDinstance'
```

- To view the action of the THD instance named myTHDinstance that is in the THD class named myTHDclass, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdInstanceAction.'myTHDclass.myTHDinstance'
```

- To set the action of the THD instance named myTHDinstance that is in the THD class named myTHDclass to resume, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlThdInstanceAction.'myTHDclass.myTHDinstance' i 2
```


Monitor and Manage System Power (SNMP)

Description	Links
Review SNMP requirements for managing system power properties	<ul style="list-style-type: none">• “Before You Begin - Power Management (SNMP)” on page 91
SMP procedures for monitoring the power consumption interfaces	<ul style="list-style-type: none">• “Monitoring the Power Consumption Interfaces (SNMP)” on page 92
SNMP procedure for maintaining the system power policy	<ul style="list-style-type: none">• “Maintaining System Power Policy (SNMP)” on page 97
SNMP procedures for applying power to the system	<ul style="list-style-type: none">• “Managing System Power Properties (SNMP)” on page 98

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, power management
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, power management
- *Oracle ILOM 3.0 Daily Management Web Procedures*, power management

Before You Begin - Power Management (SNMP)

Prior to performing the procedures in this section, you should ensure that the following requirements are met.

- Before you can use SNMP to view and configure ILOM settings, you must configure SNMP. For more information, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#).

- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user with read-write (rw) privileges.

Note – For examples of SNMP commands, see [“SNMP Command Examples” on page 263](#).

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work as presented only if you have Net-SNMP and the Net-SNMP sample applications installed.

Monitoring the Power Consumption Interfaces (SNMP)

Description	Links
SNMP procedures for monitoring the power consumption interfaces	<ul style="list-style-type: none">• “Monitor System Total Power Consumption (SNMP)” on page 93• “Monitor Actual Power Consumption (SNMP)” on page 93• “Monitor Individual Power Supply Consumption (SNMP)” on page 94• “Monitor Available Power (SNMP)” on page 96• “Monitor Hardware Configuration Maximum Power Consumption (SNMP)” on page 96• “Monitor Permitted Power Consumption (SNMP)” on page 96

Note – The power consumption interfaces described in this section might or might not be implemented on the platform that you are using. See the platform-specific ILOM supplement, platform administration guide, or product notes included with your system for implementation details.

Note – The ability to view and set the power policy is not available on SPARC platforms using ILOM 3.0 or ILOM 3.0.2. The power policy setting is available on some SPARC platforms starting with ILOM 3.0.3.

▼ Monitor System Total Power Consumption (SNMP)

- To view total system power consumption using SNMP, type:
entPhysicalName.indexnumber

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress entPhysicalName.308
```

▼ Monitor Actual Power Consumption (SNMP)

- To view actual power consumption using SNMP, type:
sunHwCtrlPowerMgmtActual.0

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress  
sunHwCtrlPowerMgmtActual.0
```

▼ Monitor Individual Power Supply Consumption (SNMP)

- To view the power consumption of an individual power supply, type `entPhysicalName` followed by the power input or power output index numbers.

For example, if you know that the `entPhysicalIndex` of `/SYS/VPS` is 303, you can view total output power consumption by typing the following command:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress \
entPhysicalName.303 \
entPhysicalClass.303 \
entPhysicalDescr.303 \
sunPlatNumericSensorBaseUnits.303 \
sunPlatNumericSensorExponent.303 \
sunPlatNumericSensorCurrent.303 \
sunPlatNumericSensorLowerThresholdNonCritical.303 \
sunPlatNumericSensorUpperThresholdNonCritical.303 \
sunPlatNumericSensorLowerThresholdCritical.303 \
sunPlatNumericSensorUpperThresholdCritical.303 \
sunPlatNumericSensorLowerThresholdFatal.303 \
sunPlatNumericSensorUpperThresholdFatal.303
```

The following table provides a brief description of each of the MIB objects included in the Power Supply Power Consumption SNMP MIB Objects command example. For more information, see the ENTITY-MIB and the SUN-PLATFORM-MIB.

MIB Object	MIB Name	Description
<code>entPhysicalName</code>	ENTITY-MIB	The textual name of the physical entity.
<code>entPhysicalClass</code>	ENTITY-MIB	The general hardware type of the physical entity.
<code>entPhysicalDescr</code>	ENTITY-MIB	A textual description of physical entity.
<code>sunPlatNumericSensorBaseUnits</code>	SUN-PLATFORM-MIB	The base unit of the values returned by this sensor as per <code>CIM_NumericSensor.BaseUnits</code> .
<code>sunPlatNumericSensorExponent</code>	SUN-PLATFORM-MIB	The exponent to be applied to the units returned by this sensor as for <code>CIM_NumericSensor.UnitModifier</code> .

MIB Object	MIB Name	Description
sunPlatNumeric SensorCurrent	SUN-PLATFORM-MIB	The sunPlatDiscreteSensorStatesIndex of a row in the sunPlatDiscreteSensorStatesTable that corresponds to the current reading of the sensor.
sunPlatNumeric SensorLower ThresholdNon Critical	SUN-PLATFORM-MIB	The lower threshold at which a non-critical condition occurs as defined for CIM_NumericSensor.LowerThresholdNonCritical.
sunPlatNumeric SensorUpper ThresholdNon Critical	SUN-PLATFORM-MIB	The upper threshold at which a non-critical condition occurs as defined for CIM_NumericSensor.UpperThresholdNonCritical.
sunPlatNumeric SensorLower ThresholdCritical	SUN-PLATFORM-MIB	The lower threshold at which a critical condition occurs as defined for CIM_NumericSensor.LowerThresholdCritical.
sunPlatNumeric SensorUpper ThresholdCritical	SUN-PLATFORM-MIB	The upper threshold at which a critical condition occurs as defined for CIM_NumericSensor.UpperThresholdCritical.
sunPlatNumeric SensorLower ThresholdFatal	SUN-PLATFORM-MIB	The lower threshold at which a fatal condition occurs as defined for CIM_NumericSensor.LowerThresholdFatal.
sunPlatNumeric SensorUpper ThresholdFatal	SUN-PLATFORM-MIB	The upper threshold at which a fatal condition occurs as defined for CIM_NumericSensor.UpperThresholdFatal.

▼ Monitor Available Power (SNMP)

- To view total available power using SNMP, type:

sunHwCtrlPowerMgmtAvailablePower.0

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress  
sunHwCtrlPowerMgmtAvailablePower.0
```

▼ Monitor Hardware Configuration Maximum Power Consumption (SNMP)

- To view the hardware configuration maximum power consumption using SNMP, type:

sunHwCtrlPowerMgmtHWConfigPower.0

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress  
sunHwCtrlPowerMgmtHWConfigPower.0
```

▼ Monitor Permitted Power Consumption (SNMP)

- To view permitted power consumption using SNMP, type:

sunHwCtrlPowerMgmtPermittedPower.0

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress  
sunHwCtrlPowerMgmtPermittedPower.0
```

▼ Monitor Power Management Properties (SNMP)

Note – You can use the `get` command to view power management settings. For a description of the MIB objects used in these commands, see the SUN-HW-CTRL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:
- ```
ssh username@snmp_manager_ipaddress
```
- Password:** *password*
2. To monitor various power consumption properties on a managed device, see the following SNMP command examples.
- To view the name of the power management policy for PowerMgmtTable index number 5, type:
- ```
sunHwCtrlPowerMgmtName.5
```

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
sunHwCtrlPowerMgmtName.5
```

- To view the units for the value of the power management policy for PowerMgmtTable index number 5, type:
- ```
sunHwCtrlPowerMgmtUnits.5
```

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
sunHwCtrlPowerMgmtUnits.5
```

- To view the value of the power management policy for PowerMgmtTable index number 5, type:
- ```
sunHwCtrlPowerMgmtValue.5
```

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
sunHwCtrlPowerMgmtValue.5
```



Maintaining System Power Policy (SNMP)

Description	Links
SNMP procedure for managing ILOM’s system power policy	<ul style="list-style-type: none">• “View and Set the Power Policy (SNMP)” on page 98

▼ View and Set the Power Policy (SNMP)

Note – You can use the `get` and `set` commands to view and set power policy. For a description of valid MIB objects for this procedure, see the table following this procedure.

1. To view the power policy using SNMP, type: `sunHwCtrlPowerMgmtPolicy.0`

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress
sunHwCtrlPowerMgmtPolicy.0
```

2. To set the power policy, use the `snmpset` command.

For example, to set this MIB object property to performance, type:
`sunHwCtrlPowerMgmtPolicy.0 i 3`

```
% snmpset -v2c -cprivate -mALL snmp_agent_ipaddress
sunHwCtrlPowerMgmtPolicy.0 i 3
```

The following table describes the System Power Policy SNMP MIB object.

MIB Object	Values	Type	Default
sunHwCtrlPowerMgmtPolicy	notsupported(1) , unknown(2) , performance(3) , elastic(4)	Integer	None

Managing System Power Properties (SNMP)

Description	Links
SNMP procedure to power on the managed interface	<ul style="list-style-type: none">• “Power On System (SNMP)” on page 99
SNMP procedure to reset the power on the managed interface	<ul style="list-style-type: none">• “Reset System Power (SNMP)” on page 99

▼ Power On System (SNMP)

Note – You can use the `set` command to configure the power setting. For a description of the MIB object used in this command, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To power on the power control target named '/SYS', type the following SNMP command

```
ilomCtrlPowerAction ./SYS' i 1
```

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlPowerAction./SYS' i 1
```

▼ Reset System Power (SNMP)

Note – You can use the `set` command to configure the reset setting. For a description of the MIB objects used in this command, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. To reset the control target named '/SP', type:

```
ilomCtrlResetAction ./SP' i 1
```

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlResetAction./SP' i 1
```


Manage Oracle ILOM Firmware Updates (SNMP)

Description	Links
Maintain Oracle ILOM firmware updates using SNMP.	<ul style="list-style-type: none">• “Update Oracle ILOM Firmware (SNMP)” on page 101

Related Information

- *Oracle ILOM 3.0 Maintenance and Diagnostics*, Oracle ILOM firmware operations
- *Oracle ILOM 3.0 Maintenance and Diagnostics*, updating Oracle ILOM firmware

▼ Update Oracle ILOM Firmware (SNMP)

Before You Begin

- Before you can use SNMP to view and configure ILOM settings, you must configure SNMP. For more information, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#).
- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user with read-write (rw) privileges.
- For examples of SNMP commands, see [“SNMP Command Examples” on page 263](#).

Note – You can use the `get` and `set` commands to view and configure ILOM firmware settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work as presented only if you have Net-SNMP and the Net-SNMP sample applications installed.

To update the Oracle ILOM firmware using SNMP, follow these steps:

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

```
Password: password
```

2. Refer to the following SNMP command examples:

- To view the version of the current firmware image, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtVersion.0
```

- To view the build number of the current firmware image, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtBuildNumber.0
```

- To view the build date and time of the current firmware image, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtBuildDate.0
```

- To view the IP address of the TFTP server that will be used to download the firmware image, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareTFTPServerIP.0
```

- To set the IP address of the TFTP server that will be used to download the firmware image, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareTFTPServerIP.0 s ipaddress
```

- To view the relative path of the new firmware image file on the TFTP server, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareTFTPFileName.0
```

- To set the relative path of the new firmware image file on the TFTP server, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareTFTPFileName.0 s 'tftpfilename'
```

- To view the property that determines whether the previous configuration of the server should be preserved after a firmware update, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwarePreserveConfig.0
```

- To set the PreserveConfig property to true so that the previous configuration of the server is preserved after a firmware update, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwarePreserveConfig.0 i 1
```

- To view the property that indicates the status of a firmware update, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtStatus.0
```

- To view the property that is used to initiate a firmware update using the values of the other firmware management properties as parameters, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtAction.0
```

- To set the property so as to initiate a firmware update using the values of the other firmware management properties as parameters, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtAction.0 i 2
```

- To clear the values of the other firmware management properties used if and when a firmware update is initiated, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtAction.0 i 1
```

- To view the version of the current firmware management file system, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareMgmtFilesystemVersion.0
```

- To view the property that is used to postpone the BIOS upgrade until the next server power off, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareDelayBIOS.0
```

- To set the DelayBIOS property to postpone the BIOS upgrade until the next server power off, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlFirmwareDelayBIOS.0 i 1
```

Manage ILOM Backup and Restore Configurations (SNMP)

Description	Links
SNMP procedure for maintaining ILOM backup and restore properties.	<ul style="list-style-type: none">• “View and Configure Backup and Restore Properties (SNMP)” on page 105

Related Information

- *Oracle ILOM 3.0 Maintenance and Diagnostics*, configuration management overview
- *Oracle ILOM 3.0 Maintenance and Diagnostics*, backing up and restoring the Oracle ILOM configuration

▼ View and Configure Backup and Restore Properties (SNMP)

Before You Begin

- Before you can use SNMP to view and configure ILOM settings, you must configure SNMP. For more information, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#).
- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user with read-write (rw) privileges.

Note – You can use the `get` and `set` commands to view and configure backup and restore settings. For a description of the MIB objects used in these commands, see the `SUN-ILOM-CONTROL-MIB`.

Note – For examples of SNMP commands, see “SNMP Command Examples” on page 263.

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work as presented only if you have Net-SNMP and the Net-SNMP sample applications installed.

To set the Oracle ILOM backup and restore properties using SNMP, follow these steps:

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
```

Password: *password*

2. Refer to the following SNMP command examples:

- To view the power policy using SNMP, type:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ipaddress  
sunHwCtrlPowerMgmtPolicy.0
```

- To configure the power property and apply it to the power control target named '/SYS', type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlPowerAction.'/SYS' i 1
```

- To restore the configuration on the SP to the original factory default state, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlResetToDefaultsAction.0 i 3
```

- To view the target destination of the configuration XML file during the backup and restore operation, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
lomCtrlBackupAndRestoreTargetURI.0
```


- To set the target destination of the configuration XML file during the backup and restore operation to
tftp://10.8.136.154/remotedir/config_backup.xml, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlBackupAndRestoreTargetURI.0 s  
'tftp://10.8.136.154/remotedir/config_backup.xml'
```

- To set the passphrase to encrypt or decrypt sensitive data during the backup and restore operation, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlBackupAndRestorePassphrase.0 s 'passphrase'
```

- To view the property used to issue an action, either backup or restore, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlBackupAndRestoreAction.0
```

- To issue a restore action using the ilomCtrlBackupAndRestoreAction MIB object, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlBackupAndRestoreAction.0 i 2
```

- To monitor the current status of the backup or restore operation, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlBackupAndRestoreActionStatus.0
```

- To specify the reset action and apply it to the reset control target named '/SP', type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlResetAction.'/SP' i 1
```


Manage SPARC Diagnostics, POST, and Boot Mode Operations (SNMP)

Description	Links
Review requirements for managing SPARC cconfiguration management interfaces	<ul style="list-style-type: none">• “Before You Begin - Manage SPARC Hosts (SNMP)” on page 109
SNMP procedures for managing SPARC management interface properties	<ul style="list-style-type: none">• “Managing SPARC Diagnostic, POST, and Boot Mode Properties (SNMP)” on page 110

Related Information

- *Oracle ILOM 3.0 Maintenance and Diagnostics*, system diagnostics overview
- *Oracle ILOM 3.0 Maintenance and Diagnostics*, SPARC diagnostics tools
- *Oracle’s Sun SPARC Enterprise Server*, diagnostic tools overview
- *Oracle’s Sun SPARC Enterprise Server*, POST overview and examples
- *Oracl’s Sun SPARC Enterprise Server*, boot mode overview

Before You Begin - Manage SPARC Hosts (SNMP)

Prior to performing the SNMP procedures for managing SPARC diagnostics, POST, and boot mode properties, you should ensure that the following requirements are met.

- Before you can use SNMP to view and configure ILOM settings, you must configure SNMP. For more information, see [“Configuring SNMP Settings in Oracle ILOM” on page 7](#).

- To execute the `snmpset` command, you need to use an SNMP v1 or v2c community or an SNMP v3 user with read-write (rw) privileges.
- For examples of SNMP commands, see [“SNMP Command Examples” on page 263](#).

Note – The example SNMP commands presented in this section are based on the Net-SNMP sample applications and, therefore, will work as presented only if you have Net-SNMP and the Net-SNMP sample applications installed.

Managing SPARC Diagnostic, POST, and Boot Mode Properties (SNMP)

Description	Links
SNMP procedures for configuring SPARC remote host diagnostic properties	<ul style="list-style-type: none"> • “Manage SPARC Host Diagnostic Properties (SNMP)” on page 110
SNMP procedures for configuring SPARC remote host control properties	<ul style="list-style-type: none"> • “Manage SPARC Host POST Operations (SNMP)” on page 113
SNMP procedures for configuring SPARC remote host boot properties	<ul style="list-style-type: none"> • “Manage SPARC Host Boot Mode Properties (SNMP)” on page 117
SNMP procedure for configuring SPARC remote host keyswitch properties	<ul style="list-style-type: none"> • “Manage SPARC Host Keyswitch Property (SNMP)” on page 118

▼ Manage SPARC Host Diagnostic Properties (SNMP)

Note – You can use the `get` and `set` commands to view and configure SPARC diagnostic settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress  
Password: password
```

2. Refer to the following SNMP command examples:

- To view the triggers of embedded diagnostics for the host, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsTrigger.0
```

- To set the triggers of embedded diagnostics for the host to power-on-reset, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsTrigger.0 i 4
```

- To view the modes for POST, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsMode.0
```

- To set the POST mode to service, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsMode.0 i 3
```

- To view the level of embedded diagnostics that should be run on the host during a boot for the power-on-reset trigger, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsPowerOnLevel.0
```

- To set the level of embedded diagnostics that should be run on the host during a boot for the power-on-reset trigger to normal, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsPowerOnLevel.0 i 3
```

- To view the level of embedded diagnostics that should be run on the host during a boot for the user-reset trigger, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsUserResetLevel.0
```

- To set the level of embedded diagnostics that should be run on the host during a boot for the user-reset trigger to normal, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsUserResetLevel.0 i 3
```

- To view the level of embedded diagnostics that should be run on the host during a boot for the error-reset trigger, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsErrorResetLevel.0
```

- To set the level of embedded diagnostics that should be run on the host during a boot for the error-reset trigger to normal, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsErrorResetLevel.0 i 3
```

- To view the verbosity level of embedded diagnostics that should be run on the host during a boot, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsPowerOnVerbosity.0
```

- To set the verbosity level of embedded diagnostics that should be run on the host during a boot to maximum, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsPowerOnVerbosity.0 i 4
```

- To view the verbosity level of embedded diagnostics that should be run on the host during a boot for user-reset trigger, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsUserResetVerbosity.0
```

- To set the verbosity level of embedded diagnostics that should be run on the host during a boot for user-reset trigger to maximum, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCDiagsUserResetVerbosity.0 i 4
```

- To view the verbosity level of embedded diagnostics that should be run on the host during a boot for error-reset trigger, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCDiagsErrorResetVerbosity.0
```

- To set the verbosity level of embedded diagnostics that should be run on the host during a boot for error-reset trigger to maximum, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCDiagsErrorResetVerbosity.0 i 4
```

- To view the progress of POST diagnostics on the host, expressed as a percentage, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCDiagsStatus.0
```

- To view the property that shows the action to control the POST diagnostics on the host, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCDiagsAction.0
```

- To set the property to take control of the POST diagnostics running on the host to start, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCDiagsAction.0 i 2
```

▼ Manage SPARC Host POST Operations (SNMP)

Note – You can use the `get` and `set` commands to view and configure SPARC host settings. For a description of the MIB objects used in these commands, see the `SUN-ILOM-CONTROL-MIB`.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the starting MAC address for the host, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostMACAddress.0
```

- To view the version string for OpenBoot PROM (OBP), type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostOBPVersion.0
```

- To view the version string for POST, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostPOSTVersion.0
```

- To view the option that determines whether the host should continue to boot in the event of a non-fatal POST error, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostAutoRunOnError.0
```

- To configure the host to continue to boot in the event of a non-fatal POST error, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostAutoRunOnError.0 i 1
```

- To view the string that describes the status of POST, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostPOSTStatus.0
```

- To view the option that determines what action the SP will take when it discovers that the host is hung, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostAutoRestartPolicy.0
```

- To configure the SP to reset when it discovers that the host is hung, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostAutoRestartPolicy.0 i 2
```


- To view the string that describes the boot status of host operating system, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostOSBootStatus.0
```

- To view the boot timer time-out value, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootTimeout.0
```

- To set the boot timer time-out value to 30 seconds, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootTimeout.0 i 30
```

- To view the property that determines what action the SP will take when the boot timer expires, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootRestart.0
```

- To configure the SP to reset when the boot timer expires, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootRestart.0 i 2
```

- To view the maximum number of boot failures allowed by the SP, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostMaxBootFail.0
```

- To set the maximum number of boot failures allowed by the SP to 10, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostMaxBootFail.0 i 10
```

- To view the property that determines what action the SP will take when the maximum number of boot failures is reached, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootFailRecovery.0
```

- To configure the SP to power cycle the host when the maximum number of boot failures is reached, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostBootFailRecovery.0 i 2
```

- To view the version string for the Hypervisor, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostHypervisorVersion.0
```

- To view the version string for the system firmware (SysFw), type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostSysFwVersion.0
```

- To view the property that determines the break action that SP will send, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostSendBreakAction.0
```

- To configure the SP to send a `dumpcore` break action, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostSendBreakAction.0 i 3
```

- To view the property that determines the host I/O reconfiguration policy to apply on next host power-on, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostIoReconfigurePolicy.0
```

- To configure the SP to execute the host I/O reconfiguration policy on the next power-on, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCHostIoReconfigurePolicy.0 i 3
```

▼ Manage SPARC Host Boot Mode Properties (SNMP)

Note – You can use the `get` and `set` commands to view and configure SPARC boot mode settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress
Password: password
```

2. Refer to the following SNMP command examples:

- To view the boot mode state for the host, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeState.0
```

- To configure the host to retain current NVRAM variable settings, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeState.0 i 1
```

- To view the boot script to use when the boot mode state is set to `script`, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeScript.0
```

- To specify the boot script to use when the boot mode state is set to `'setenv diag-switch'`, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeScript.0 s 'setenv diag-switch'
```

- To view date and time when the boot mode configuration will expire, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeExpires.0
```

- To view the string that refers to the LDOM configuration name, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress
ilomCtrlSPARCBootModeLDMConfig.0
```

- To set the LDOM configuration name to default, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCBootModeLDOMConfig.0 s default
```

▼ Manage SPARC Host Keyswitch Property (SNMP)

Note – You can use the `get` and `set` commands to view and configure SPARC key switch settings. For a description of the MIB objects used in these commands, see the SUN-ILOM-CONTROL-MIB.

1. Log in to a host that has an SNMP tool and the ILOM MIBs installed. For example, type:

```
ssh username@snmp_manager_ipaddress  
Password: password
```

2. Refer to the following SNMP command examples:

- To view the current state of the virtual key switch, type:

```
% snmpget -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCKeySwitchState.0
```

- To set the state of the virtual key switch to standby, type:

```
% snmpset -mALL -v2c -cprivate SNMP_agent_ipaddress  
ilomCtrlSPARCKeySwitchState.0 i 2
```

Server Managment Using IPMI

Description	Links
Learn about using IPMITool to manage Oracle servers	<ul style="list-style-type: none">• “Intelligent Platform Management Interface (IPMI)” on page 119• “About IPMI” on page 120• “IPMITool” on page 121• “IPMI Alerts” on page 121• “IPMI Administrator and Operator Roles” on page 122
Learn how to configure the IPMI state and perform various mangement functions using the IPMITool	<ul style="list-style-type: none">• “Configuring the IPMI State” on page 122• “Using IPMITool to Run ILOM CLI Commands” on page 123• “Performing System Management Tasks (IPMITool)” on page 126
Learn about the IPMI commands	<ul style="list-style-type: none">• “IPMITool Utility and Command Summary” on page 137

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, user management
- *Oracle ILOM 3.0 Daily Management Concepts*, alert management
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, user management
- *Oracle ILOM 3.0 Daily Management CLI Procedures*, CLI overview
- *Oracle ILOM 3.0 Daily Management Web Procedures*, user management

Intelligent Platform Management Interface (IPMI)

- [“About IPMI” on page 120](#)

- “IPMItool” on page 121
- “IPMI Alerts” on page 121
- “IPMI Administrator and Operator Roles” on page 122

About IPMI

ILOM supports the Intelligent Platform Management Interface (IPMI), which enables you to monitor and control your server platform, as well as to retrieve information about your server platform.

IPMI is an open, industry-standard interface that was designed for the management of server systems over a number of different types of networks. IPMI functionality includes field-replaceable unit (FRU) inventory reporting, system monitoring, logging of system events, system recovery (including system resets and power-on and power-off capabilities), and alerting.

The monitoring, logging, system recovery, and alerting functions available through IPMI provide access to the manageability that is built into the platform hardware.

ILOM is compliant with IPMI v1.5 and v2.0.

An Oracle-provided Windows port of IPMItool is available at:

<http://www.oracle.com/technetwork/server-storage/servermgmt/downloads/index.html>

Additional information, including detailed specifications about IPMI, is available at the following sites:

- <http://www.intel.com/design/servers/ipmi/spec.htm>
- <http://openipmi.sourceforge.net>

The service processors (SPs) on your servers and server modules (blades) are IPMI v2.0 compliant. You can access IPMI functionality through the command line using the `IPMItool` utility either in-band (using the host operating system running on the server) or out-of-band (using a remote system). Additionally, you can generate IPMI-specific traps from the ILOM web interface, or manage the SP's IPMI functions from any external management solution that is IPMI v1.5 or v2.0 compliant.

IPMItool

IPMItool is an open-source, simple command-line interface (CLI) utility for managing and configuring IPMI-enabled devices. IPMItool can be used to manage the IPMI functions of either the local system or a remote system. You can use the IPMItool utility to perform IPMI functions with a kernel device driver or over a LAN interface. You can download IPMItool from this site:

<http://ipmitool.sourceforge.net/>

You can do the following with IPMItool:

- Read the Sensor Data Record (SDR) repository.
- Print sensor values.
- Display the contents of the system event log (SEL).
- Print field-replaceable unit (FRU) inventory information.
- Read and set LAN configuration parameters.
- Perform remote chassis power control.

Detailed information about IPMItool is provided in a man page that is available from this site:

<http://ipmitool.sourceforge.net/manpage.html>

IPMItool supports a feature that enables you to enter ILOM command-line interface (CLI) commands just as though you were using the ILOM CLI directly. CLI commands can be scripted, and then the script can be run on multiple service processor (SP) instances.

IPMI Alerts

ILOM supports alerts in the form of IPMI Platform Event Trap (PET) alerts. Alerts provide advance warning of possible system failures. Alert configuration is available from the ILOM SP on your server or server module. IPMI PET alerts are supported on all Oracle Sun server platforms and modules, with the exception of the chassis monitoring module (CMM). For more information about the types of IPMI alerts, refer to “Alert Management” in the *Oracle ILOM 3.0 Daily Management Concepts Guide*.

IPMI Administrator and Operator Roles

The *IPMI Administrator* role maps to these user roles in ILOM: `aucro`. The *IPMI Operator* role maps to these user roles in ILOM: `cro`. A brief explanation of these ILOM roles appears in the following table.

TABLE: IPMI Administrator and Operator Roles in ILOM

IPMI Role	Enabled ILOM Role Privileges	Description
Administrator	<ul style="list-style-type: none">• Admin (a)• User Management (u)• Console (c)• Reset and Host Console (r)• Read-Only (o)	These user roles enable read and write privileges to these management features in ILOM: system management configuration properties, user account properties, remote console management properties, remote power management properties, and reset and host control management properties.
Operator	<ul style="list-style-type: none">• Console (c)• Reset and Host Console (r)• Read-Only (o)	These user roles enable read and write privileges to these management features in ILOM: remote console management properties, remote power management properties, and reset and host control management properties. The Read-Only role also provides read access to system management configuration properties and user management properties.

For more information about ILOM roles and privileges, refer to “User Management” in the *Oracle ILOM 3.0 Daily Management Concepts Guide*.

Configuring the IPMI State

Description	Links
ILOM CLI procedure to enable the IPMI state	“Enable IPMI State (CLI)” on page 123
ILOM web interface procedure to enable the IPMI state	“Enable IPMI State (Web)” on page 123

▼ Enable IPMI State (CLI)

1. Log in to the ILOM CLI. using an account with IPMI Administrator privileges.

For more information about enabling IPMi administrator privileges, see [“IPMI Administrator and Operator Roles” on page 122.](#)

2. At the command prompt, type:

-> **set /SP/services/ipmi servicestate=enabled**

For example:

```
-> set /SP/services/ipmi servicestate=enabled
Set 'servicestate' to 'enabled'
```

▼ Enable IPMI State (Web)

1. Log in to the ILOM web interface using an account with IPMI administrator privileges.

For more information about enabling IPMi administrator privileges, see [“IPMI Administrator and Operator Roles” on page 122.](#)

2. Click Configuration --> System Management Access --> IPMI.

The IPMI Settings page appears.

3. Click the check box to enable or disable the IPMI state.

Using IPMItool to Run ILOM CLI Commands

The IPMItool CLI is a convenient alternative method to executing ILOM CLI commands. It enables you to enter ILOM CLI commands just as if you were using the ILOM CLI directly. Most ILOM CLI commands are supported.

Description	Links
Requirements for using IPMItool to run ILOM CLI commands	“Before You Begin - IPMItool and ILOM Requirements” on page 124
Procedure for enabling you to use the IPMItool to run CLI commands	“Access the ILOM CLI From IPMItool” on page 124
Create and run ILOM CLI command scripts	“Scripting ILOM CLI Commands With IPMItool” on page 125

Before You Begin - IPMItool and ILOM Requirements

- To use the ILOM CLI through IPMItool, you must be using IPMItool version 1.8.9.4 or later. To check the version number of IPMItool, type:
ipmitool -V
- Ensure that you have the proper user roles assigned in ILOM when using the IPMItool CLI to execute ILOM commands. For more information, see [“IPMI Administrator and Operator Roles” on page 122](#).

▼ Access the ILOM CLI From IPMItool

1. To enable the ILOM CLI using IPMItool, type:

```
# ipmitool -H hostname -U username -P userpassword sunoem cli
```

The ILOM CLI prompt appears as follows:

```
Connected. Use ^D to exit.
->
```

2. To use the CLI, type CLI commands.

To script ILOM CLI commands, see [“Scripting ILOM CLI Commands With IPMItool” on page 125](#).

Scripting ILOM CLI Commands With IPMITool

A key benefit of using ILOM CLI from IPMITool is that the CLI commands can be scripted and then the script can be run on multiple SP instances. Scripting is possible because the CLI commands can be included on the IPMITool command line where each argument on the command line is treated as a separate ILOM CLI command. Command separation is archived by including quotation marks at the beginning and end of each ILOM CLI command.

The following example shows how to include two CLI commands on the IPMITool command line. In the example, notice that each ILOM CLI command begins and ends with quotation marks.

```
# ipmitool -H hostname -U username -P userpassword sunoem cli "show
/SP/services" "show /SP/logs"
Connected. Use ^D to exit.
-> show /SP/services
/SP/services
Targets:
-> show /SP/logs
http
https
/SP/logs
->Session closed
servicetag
Disconnected
Targets:
snmp
event
ssh
Properties:
sso
Properties:
Commands:
Commands:
cd
cd
show
show
```



Performing System Management Tasks (IPMItool)

Description	Links
Review ILOM user access requirements	<ul style="list-style-type: none">• “Before You Begin - ILOM and IPMItool Requirements” on page 126
Monitor sensors and values using IPMItool	<ul style="list-style-type: none">• “Display Sensor List (IPMItool)” on page 127• “View Single Sensor Details (IPMItool)” on page 128• “View and Interpret Presence Sensor Type Values” on page 128
Remotely manage host power using IPMItool	<ul style="list-style-type: none">• “Power On Host (IPMItool)” on page 130• “Power Off Host (IPMItool)” on page 130• “Power Cycle Host (IPMItool)” on page 130• “Shut Down Host Gracefully (IPMItool)” on page 131
Manage power usage using IPMItool	<ul style="list-style-type: none">• “Manage ILOM Power Budget Interfaces (IPMItool)” on page 131
Identify field replacement unit manufacturing information	<ul style="list-style-type: none">• “Display FRU Manufacturing Details (IPMItool)” on page 135
Monitor the system event log using IPMItool	<ul style="list-style-type: none">• “Display ILOM Event Log Using IPMItool” on page 136

Before You Begin - ILOM and IPMItool Requirements

Ensure that you have the proper user roles assigned in ILOM when using the IPMItool CLI to execute ILOM commands. For more information, see [“IPMI Administrator and Operator Roles” on page 122](#).

▼ Display Sensor List (IPMItool)

- To view a list of sensors on a managed device, type:

sdr list

For example:

\$ ipmitool -H 1.2.3.4 -I lanplus -U username -P userpassword sdr list		
/SYS/T_AMB	24 degrees C	ok
/RFM0/FAN1_SPEED	7110 RPM	ok
/RFM0/FAN2_SPEED	5880 RPM	ok
/RFM1/FAN1_SPEED	5880 RPM	ok
/RFM1/FAN2_SPEED	6360 RPM	ok
/RFM2/FAN1_SPEED	5610 RPM	ok
/RFM2/FAN2_SPEED	6510 RPM	ok
/RFM3/FAN1_SPEED	6000 RPM	ok
/RFM3/FAN2_SPEED	7110 RPM	ok
/RFM4/FAN1_SPEED	6360 RPM	ok
/RFM4/FAN2_SPEED	5610 RPM	ok
/RFM5/FAN1_SPEED	5640 RPM	ok
/RFM5/FAN2_SPEED	6510 RPM	ok
/RFM6/FAN1_SPEED	6180 RPM	ok
/RFM6/FAN2_SPEED	6000 RPM	ok
/RFM7/FAN1_SPEED	6330 RPM	ok
/RFM7/FAN2_SPEED	6330 RPM	ok
/RFM8/FAN1_SPEED	6510 RPM	ok
/RFM8/FAN2_SPEED	5610 RPM	ok

Note – If `bimetal` is not configured to support the `-P` option, which enables the password to be entered in the command line, you will be prompted to enter the password.

Note – The example sensor output shown in the preceding example was shortened. The actual output displays 163 sensors.

▼ View Single Sensor Details (IPMItool)

- To view details about a single sensor on a managed device, type:

```
sensor get /target/sensor_name
```

For example, to view sensor details about the system temperature (/SYS/T_AMB), you would type:

```
sensor get /SYS/T_AMB
```

/SYS/T_AMB example output:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword sensor get /SYS/T_AMB
Locating sensor record...
Sensor ID           : /SYS/T_AMB (0x8)
Entity ID           : 41.0
Sensor Type (Analog) : Temperature
Sensor Reading       : 24 (+/- 0) degrees C
Status               : ok
Lower Non-Recoverable : 0.000
Lower Critical        : 4.000
Lower Non-Critical    : 10.000
Upper Non-Critical    : 35.000
Upper Critical        : 40.000
Upper Non-Recoverable : 45.000
Assertions Enabled    : lnc- lcr- lnr- unc+ ucr+ unr+
Deassertions Enabled  : lnc- lcr- lnr- unc+ ucr+ unr+
```

▼ View and Interpret Presence Sensor Type Values

Before You Begin

- The IPMItool supports the output of a States Asserted field for each presence sensor type record. This States Asserted field can appear in the IPMItool output as either:

- States Asserted = Entity Presence

When the States Asserted = Entity Presence field appears, the sensor output for a hardware component can show one of three valid values:

Present(=1), Absent(=2), Disabled(=4).

- or -

- States Asserted =Availability State

When the States Asserted = Availability State field appears, the sensor output for a hardware component can show one of two valid values:

Device Absent(=1) and Device Present(=2).

Note – Oracle ILOM supports the output of both `States Asserted` fields. However, some Oracle hardware platforms might support both or one of the possible `States Asserted` fields (`Entity Presence` or `Availability State`).

For additional information about how to interpret values presented for IPMI presence sensor types, refer to Section 42 - Sensor and Event Code Tables in the IPMI 2.0 Specifications. Understanding all of Section 42 is critical in understanding how to interpret a sensor value.

For further information about sensor details that are specific to an Oracle hardware platform, see the platform Oracle ILOM supplement guide or administration guide.

To view and interpret IPMITool presense sensor type values, follow these steps:

- 1. **To view the actual sensor reading for hardware components, use the IPMITool `sdr list` command.**

For example, after issuing the `sdr list` command the following presence sensor type readings appear for PCIE hardware components.

PCIE_CC/PRSNT	0x02	ok
PCIE0/F20/PRSNT	0x01	ok

- 2. **To determine the `States Asserted` field value for a presence sensor type, use the IPMITool `sensor get` command.**

One of the following `States Asserted` fields appear after issuing the `sensor get` command from the IPMITool:

- `States Asserted = Entity Presence`

In the following example, the value shown for the `States Asserted = Entity Presence` field is *Absent*.

```
$ ipmitool sensor get PCIE_CC/PRSNT
Locating sensor record...
Sensor ID           : PCIE_CC/PRSNT (0xad)
Entity ID           : 49.0
Sensor Type (Discrete): Entity Presence
States Asserted      : Entity Presence
[Absent]
```

- `States Asserted = Availability State`

In the following example, the value shown for the States Asserted = Availability State field is *Device Absent*.

```
$ ipmitool sensor get PCIE0/F20/PRSNT
Locating sensor record...
Sensor ID           : PCIE0/F20/PRSNT (0xe6)
Entity ID           : 11.0
Sensor Type (Discrete): Entity Presence
States Asserted      : Availability State
[Device Absent]
```

▼ Power On Host (IPMITool)

- To power on the host on a managed device, type:

chassis power on

For example:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword chassis
power on
```

▼ Power Off Host (IPMITool)

- To power off the host on a managed device, type:

chassis power off

For example:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword chassis
power off
```

▼ Power Cycle Host (IPMITool)

- To power cycle the host on a managed device, type:

chassis power cycle

For example:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword chassis
power cycle
```


▼ Shut Down Host Gracefully (IPMITool)

- To shut down the host on a managed device gracefully, type:

```
chassis power soft
```

For example:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword chassis  
power soft
```

▼ Manage ILOM Power Budget Interfaces (IPMITool)

1. To set the Power Limit Activation State on a managed device, use one of the following commands:

- To activate:

```
$ ipmitool -H <localhost|IP address> -U <username> -P <password> raw  
0x2e 0x49 0x00 0x01 0xFF 0xFF
```

Upon command completion:

```
dc
```

- To deactivate:

```
$ ipmitool -H <localhost|IP address> -U <username> -P <password> raw  
0x2e 0x49 0x00 0x00 0xFF 0xFF
```

Upon command completion:

```
dc
```

The following table describes the Power Limit Activation State (IPMITool) input and output fields:

Fields	Byte	Description
Input Data	1	Sun OEM command group number "0x2e".
	2	Command code "0x49" sets the power limit activation state.
	3	Group extension identification "0x00". The value for this field is ignored.
	4	Sub-commands for power-limit activation: 0x00 - Deactivate power limit 0x01 - Activate power limit
	5-6	Reserved fields: 0xFF. The values for this field are ignored.
Output Data	1	Completion code consumed by IPMITool. The system does not display a status for successful completion code. However, if the result of the completion code is anything other than 'successful', a failure message appears.
	2	Group extension identification "-dc" appears upon command completion.

2. To get Power Limit budget properties, use the following command:

Note – You should use a Get Power Limit Budget Wattage command prior to setting the power-limit budget wattage property.

```
$ ipmitool -H <localhost|IP address> -U <username> -P <password> raw 0x2e
0x4A 0x00 0x00 0x00
```

Upon command completion:

```
dc 01 b3 00 02 fa 00 00 00 00 01 e9 00 00
```

The following table describes the Get Power Limit (IPMITool) input and output fields:

Field	Byte	Description
Input Data	1	SUN OEM command group number 0x2e.
	2	Command code 0x4A gets Power Budget settings
	3	Group extension identification: 0x00. The value for this field is ignored.
	4-5	Reserved fields: 0x00. Values for this field are ignored.
Output Data	1	Completion Code, consumed by IPMITool. Not displayed upon command completion. However if completion code is anything other than success, then a failure message is displayed upon command completion.
	2	Group Extension Identification. Displayed as dc in the preceding example.
	3	Activation State: 00 - deactivated; 01 - activated.
	4	Reserved field. Note that the value b3 in the preceding example can be ignored.
	5	Exception action, taken if power limit is exceeded and cannot be controlled within the correction time limit. Return values: 00 - none; 01 - hard power-off.
	6-7	Power limit in watts. 02 fa in the preceding example.
	8-11	Correction timelimit in milliseconds. 00 00 00 00 in the preceding example.
	12	Flag indicating whether the correction time limit is the system default timelimit. (00 - not default; 01 - default)
	13	Reserved field. Note that the value shown (e9) in the preceding example can be ignored.
	14-15	Reserved fields. Note that the value shown (00 00) in the preceding example can be ignored.

3. To set the Power Limit, use the following command:

Note – The set power limit commands sets the power budget limit for the system. Use this command to set the maximum system power usage. The power limit should always be persistent across AC and DC cycles.

```
$ ipmitool -H <localhost|IP address> -U <username> -P <password> raw
0x2e 0x4B 0xdc 0xff 0xff 0x01 0x02 0xaa 0x00 0x00 0x1b 0x58
0x00 0xff 0x00 0x00
```

Upon command completion:

```
dc
```

The following table describes Set Power Limit (IPMITool) input and output fields:

Fields	Byte	Description
Input Data	1	SUNOEM command group number: 0x2e.
	2	Command code 0x4B sets power budget settings.
	3	Group extension identification: 0xdc . The value for this field is ignored.
	4-6	Reserved fields: 0xff 0xff 0xff. The values for this field are ignored.
	7	Exception action taken: 00 - none 01 - hard power-off
	8-9	Power limit in watts. For example: 0x2a 0xaa

Fields	Byte	Description
	10-13	Correction time limit in milliseconds. For example: 0x00 0x00 0x1b 0x58. This value is ignored if the time limit is set to default; see next byte.
	14	A flag indicating whether to use the system default time limit. Correction time limit in bytes 10-13 will be ignored. 0x00 - not default 0x01 - default
	15	Reserved field: 0xff. The value for this field is ignored.
	16-17	Reserved field: 0x00 0x00. The value for this field is ignored.
Output Data	1	Completion code that is consumed by IPMItool. The system does not display a status for successful completion code. However, if the result of the completion code is anything other than successful, a failure message appears.
	2	Group extension identification dc appears upon command completion.

▼ Display FRU Manufacturing Details (IPMItool)

- To display FRU manufacturing details on a managed device, use the **fru print** command.

For example:

```
$ ipmitool -H 1.2.3.4 -v -I lanplus -U username -P userpassword fru print
FRU Device Description : Builtin FRU Device (ID 0)
  Board Product       : ASSY,ANDY,4SKT_PCI-E,BLADE
  Board Serial        : 0000000-7001
  Board Part Number   : 501-7738-01
  Board Extra         : AXX_RevE_Blade
  Product Manufacturer : ORACLE
  Product Name        : ILOM

FRU Device Description : /SYS (ID 4)
  Chassis Type        : Rack Mount Chassis
  Chassis Part Number : 541-0251-05
  Chassis Serial      : 00:03:BA:CD:59:6F
  Board Product       : ASSY,ANDY,4SKT_PCI-E,BLADE
  Board Serial        : 0000000-7001
  Board Part Number   : 501-7738-01
```

```

Board Extra      : AXX_RevE_Blade
Product Manufacturer : ORACLE
Product Name      : SUN BLADE X8400 SERVER MODULE
Product Part Number : 602-0000-00
Product Serial    : 0000000000
Product Extra     : 080020ffffffffffffffff0003baf15c5a

```

```

FRU Device Description : /P0 (ID 5)
Product Manufacturer   : ADVANCED MICRO DEVICES
Product Part Number    : 0F21
Product Version       : 2

```

```

FRU Device Description : /P0/D0 (ID 6)
Product Manufacturer   : MICRON TECHNOLOGY
Product Name          : 1024MB DDR 400 (PC3200) ECC
Product Part Number    : 18VDDF12872Y-40BD3
Product Version       : 0300
Product Serial        : D50209DA
Product Extra         : 0190
Product Extra         : 0400

```

```

FRU Device Description : /P0/D1 (ID 7)
Product Manufacturer   : MICRON TECHNOLOGY
Product Name          : 1024MB DDR 400 (PC3200) ECC
Product Part Number    : 18VDDF12872Y-40BD3
Product Version       : 0300
Product Serial        : D50209DE
Product Extra         : 0190
Product Extra         : 0400

```

▼ Display ILOM Event Log Using IPMItool

- To view the ILOM event log on a managed device, use the **sel list** command.

For example:

```

$ ipmitool -H 1.2.3.4 -I lanplus -U username -P userpassword sel list
100 | Pre-Init Time-stamp | Power Unit #0x78 | State Deasserted
200 | Pre-Init Time-stamp | Power Supply #0xa2 | Predictive Failure Asserted
300 | Pre-Init Time-stamp | Power Supply #0xba | Predictive Failure Asserted
400 | Pre-Init Time-stamp | Power Supply #0xc0 | Predictive Failure Asserted
500 | Pre-Init Time-stamp | Power Supply #0xb4 | Predictive Failure Asserted
600 | 04/05/2007 | 12:03:24 | Power Supply #0xa3 | Predictive Failure Deasserted
700 | 04/05/2007 | 12:03:25 | Power Supply #0xaa | Predictive Failure Deasserted
800 | 04/05/2007 | 12:03:25 | Power Supply #0xbc | Predictive Failure Deasserted
900 | 04/05/2007 | 12:03:26 | Power Supply #0xa2 | Predictive Failure Asserted
a00 | 04/05/2007 | 12:03:26 | Power Supply #0xa8 | Predictive Failure Deasserted

```

b00	04/05/2007	12:03:26	Power Supply #0xb6	Predictive Failure Deasserted
c00	04/05/2007	12:03:26	Power Supply #0xbb	Predictive Failure Deasserted
d00	04/05/2007	12:03:26	Power Supply #0xc2	Predictive Failure Deasserted
e00	04/05/2007	12:03:27	Power Supply #0xb0	Predictive Failure Deasserted
f00	04/05/2007	12:03:27	Power Supply #0xb5	Predictive Failure Deasserted
1000	04/05/2007	12:03:27	Power Supply #0xba	Predictive Failure Asserted
1100	04/05/2007	12:03:27	Power Supply #0xc0	Predictive Failure Asserted
1200	04/05/2007	12:03:28	Power Supply #0xa9	Predictive Failure Deasserted
1300	04/05/2007	12:03:28	Power Supply #0xae	Predictive Failure Deasserted
1400	04/05/2007	12:03:28	Power Supply #0xb4	Predictive Failure Asserted
1500	04/05/2007	12:03:28	Power Supply #0xbe	Predictive Failure Deasserted

IPMItool Utility and Command Summary

You can download the IPMItool utility at:

<http://ipmitool.sourceforge.net/>

After you install the IPMItool package, you can access detailed information about command usage and syntax from the man page that is installed. The following table summarizes available IPMItool commands.

TABLE: IPMItool commands

IPMI Command	Function
sunoem sshkey set	Configure an SSH key for a remote shell user.
ipmitool sunoem sshkey del	Remove an SSH key from a remote shell user.
ipmitool sunoem led get	Read LED status.
ipmitool sunoem led set	Set LED status.
ipmitool sunoem cli	Enter ILOM CLI commands as if you were using the ILOM CLI directly. The LAN/LANplus interface should be used.
ipmitool sunoem CLI force	Available as of ILOM 3.0.10, a force option can be invoked as an argument to the sunoem CLI command.
ipmitool raw	Execute raw IPMI commands.
ipmitool lan print	Print the current configuration for the given channel.
ipmitool lan set (1) (2)	Set the given parameter on the given channel.

TABLE: IPMItool commands

IPMI Command	Function
<code>ipmitool chassis status</code>	Display information regarding the high-level status of the system chassis and main power subsystem.
<code>ipmitool chassis power</code>	Perform a chassis control command to view and change the power state.
<code>ipmitool chassis identify</code>	Control the front panel identify light. Default is 15. Use 0 to turn off.
<code>ipmitool chassis restart_cause</code>	Query the chassis for the cause of the last system restart.
<code>ipmitool chassis bootdev (1)</code>	Request the system to boot from an alternate boot device on next reboot.
<code>ipmitool chassis bootparam (1)</code>	Set the host boot parameters.
<code>ipmitool chassis selftest</code>	Display the BMC self-test results.
<code>ipmitool power</code>	Return the BMC self-test results.
<code>ipmitool event</code>	Send a predefined event to the system event log.
<code>ipmitool sdr</code>	Query the BMC for sensor data records (SDR) and extract sensor information of a given type, then query each sensor and print its name, reading, and status.
<code>ipmitool sensor</code>	List sensors and thresholds in a wide table format.
<code>ipmitool fru print</code>	Read all field-replaceable unit (FRU) inventory data and extract such information as serial number, part number, asset tags, and short strings describing the chassis, board, or product.
<code>ipmitool sel</code>	View the ILOM SP system event log (SEL).
<code>ipmitool pef info</code>	Query the BMC and print information about the PEF-supported features.
<code>ipmitool pef status</code>	Print the current PEF status (the last SEL entry processed by the BMC, and so on).
<code>ipmitool pef list</code>	Print the current PEF status (the last SEL entry processed by the BMC, and so on).
<code>ipmitool user</code>	Display a summary of user ID information, including maximum number of user IDs, the number of enabled users, and the number of fixed names defined.
<code>ipmitool session</code>	Get information about the specified sessions. You can identify sessions by their ID, by their handle number, by their active status, or by using the keyword “all” to specify all sessions.

TABLE: IPMItool commands

IPMI Command	Function
<code>ipmitool firewall (1)</code>	Enable or disable individual command and command sub-functions; determine which commands and command sub-functions can be configured on a given implementation.
<code>ipmitool set (1)</code>	Set the runtime options including session host name, user name, password, and privilege level.
<code>ipmitool exec</code>	Execute IPMItool commands from file name. Each line is a complete command.

Server Management Using WS-Management and CIM

Description	Links
Learn about support for WS-Management and CIM	<ul style="list-style-type: none">• “WS-Management and CIM Overview” on page 141
Learn how to configure the state for WS-Management	<ul style="list-style-type: none">• “Configuring Support for WS-Management in ILOM” on page 143
Learn about the supported CIM profiles and Oracle’s Sun specific classes	<ul style="list-style-type: none">• “Supported DMTF SMASH Profiles, CIM Classes and CIM Indications” on page 147

Related Information

- *Oracle ILOM 3.0 Daily Management Concepts*, Oracle ILOM overview

WS-Management and CIM Overview

As of version 3.0.8, ILOM supports the use of the Distributed Management Task Force (DMTF) Web Services for Management (WS-Management) protocol and Common Information Model (CIM). The support for these DMTF standards in ILOM enables developers to build and deploy network management applications to monitor and manage information about Oracle’s Sun system hardware.

Topics described in this section, include:

- [“WS-Management” on page 142](#)
- [“Common Information Model \(CIM\)” on page 142](#)

WS-Management

WS-Management is based on the Simple Object Access Protocol (SOAP) specification that promotes interoperability between managed applications and managed resources. It enables you to:

- Discover the presence of management resources, as well as provide navigation among them.
- View and write to individual management resources, such as settings and dynamic values.
- Obtain a list for contents of containers and collections, such as system components and log entries.
- Run management methods.

For further details about implementing and deploying a WS-Management environment to remotely manage system hardware across your IT infrastructure, see: <http://www.dmtf.org/standards/wsman>

For more information about how to configure support for WS-Management in ILOM, see “Configuring Support for WS-Management in ILOM” on page 143.

Common Information Model (CIM)

CIM is an object-oriented information model that provides a common definition for managing system hardware data. These common definitions enable you to exchange semantically rich management information among systems on your network.

CIM supplies a set of classes that provide a framework to organize the information about the managed environment. Specifically, these classes enable you to create or use another application other than ILOM to monitor and manage Oracle’s Sun hardware.

System Management Architecture for Server Management (SMASH)

Oracle’s Sun hardware supports a relevant subset of SMASH profiles. For more information about DMTF SMASH profiles, consult the specification for this standard at: <http://www.dmtf.org/standards/mgmt/smash>

For more information about support SMASH profiles and CIM classes, see “Supported DMTF SMASH Profiles, CIM Classes and CIM Indications” on page 147.

Configuring Support for WS-Management in ILOM

The following sections describe the prerequisites and procedures for configuring support for WS-Management in ILOM.

- “Before You Begin - WS-Management Requirements” on page 143
- “Edit the WS-Management Service State, Transport Mode, and Port Number (CLI)” on page 143
- “Edit WS-Management State, Transport Mode, and Port Number (Web)” on page 146

Before You Begin - WS-Management Requirements

To edit the configuration properties for WS-Management in ILOM, you must have Admin (a) role privileges.

▼ Edit the WS-Management Service State, Transport Mode, and Port Number (CLI)

1. Log in to the ILOM SP CLI.

Note – Alternatively, you can log in to the ILOM CMM CLI then navigate to the SP target where you want to enable or disable the KVMS lock option for the ILOM Remote Console.

2. To view all the properties associated with the management of the SP WS-Management service, type:

-> **help /SP/services/wsman**

The following help output appears for the WS-Management service:

```
/SP/services/wsman : Management of the WSMAN service
Targets:

Properties:
  http_port : WSMAN http port
  http_port : User role required for set = a
```

```
https_port : WSMAN https port
https_port : User role required for set = a

mode : WSMAN mode
mode : User role required for set = a

state : WSMAN state
state : User role required for set = a
```

3. To navigate and manage the SP WS-Management target properties, perform the tasks described in the following table.

Task	Instructions
Navigate to the WS-Management service target.	<p>To navigate to the WS-Management service target, type the following command:</p> <pre>-> cd /SP/services/wsman</pre> <p>Note - You must navigate to the wsman target prior to viewing or configuring the properties associated with the WS-Management service.</p>
View the WS-Management CLI properties and commands.	<p>To view the WS-Management properties and commands, type the following command:</p> <pre>-> show</pre> <p>Show output example:</p> <pre>-> cd /SP/services/wsman /SP/services/wsman -> show /SP/services/wsman Targets: Properties: http_port = 7783 https_port = 7782 mode = http state = enabled Commands: cd set show</pre>

Task	Instructions
Set the WS-Management service state.	<p>To enable or disable support for the WS-Management service in ILOM, type the following command to set the service state:</p> <pre>-> set state=enabled</pre> <p>or</p> <pre>-> set state=disabled</pre> <p>Note - The service state for WS-Management in ILOM 3.0.8 is, by default, disabled. For all other ILOM versions, the service state is, by default, enabled.</p>
Set the WS-Management transport mode (HTTP or HTTPS).	<p>To set the transport mode (HTTP or HTTPS) for the WS-Management service in ILOM, type one of the following commands:</p> <pre>-> set mode=http</pre> <p>or</p> <pre>-> set mode=https</pre>
Set the WS-Management transport mode port number.	<p>To set the transport mode port number for the WS-Management service in ILOM, type one of the following commands:</p> <pre>-> set http_port=####</pre> <p>or</p> <pre>-> set https_port=####</pre> <p>where #### equals the port number to be assigned to the specified transport mode (HTTP or HTTPS).</p> <p>For example, to set the default port number for HTTP or HTTPS, you would type:</p> <p>For HTTP: <code>set http_port=8889</code></p> <p>For HTTPS: <code>set https_port=8888</code></p>

4. Type `exit` to exit the ILOM CLI.

▼ Edit WS-Management State, Transport Mode, and Port Number (Web)

1. Log in to the ILOM SP web interface.
2. In the ILOM SP web interface, click Configuration --> System Management Access --> WS-Man.
3. In the WS-Man page, configure the following WS-Man settings:

Settings	Instructions
Enable or disable the WS-Management service state.	<ul style="list-style-type: none"> Click to select (enable) or clear (disable) the State Enabled check box. By default, this setting is disabled in ILOM.
Select a WS-Management transport mode (HTTP or HTTPS).	<ul style="list-style-type: none"> Click to select HTTP or HTTPS in the Mode list box. By default, this setting is set to HTTP.
Set the WS-Management transport mode port number.	<ul style="list-style-type: none"> In the HTTP or HTTPS text field, specify the transport mode port number for the WS-Management service. The default port number settings for HTTP or HTTPS are as follows: <ul style="list-style-type: none"> HTTP: 8889 HTTPS: 8888

4. Click **Save** to apply the changes made to the WS-Man settings.

Supported DMTF SMASH Profiles, CIM Classes and CIM Indications

Oracle-supported CIM classes provide a common information model interface for developers building management applications. With Oracle-specific CIM class properties, developers can use standards-based CIM-compliant applications to manage Oracle's Sun hardware.

Note – Oracle supports CIM schema version 2.18.1. For DMTF CIM schema details, see http://www.dmtf.org/standards/cim/cim_schema_v2181.

Note – Use name space (<http://schemas.oracle.com/wbem/wscim/1/cim-schema/2>) when using Oracle-specific CIM class. For example:
http://schemas.oracle.com/wbem/wscim/1/cim-schema/2/Oracle_ComputerSystem

Note – As of ILOM 3.0.14, the Oracle Sun-supported CIM classes have been renamed from Sun_xxx to Oracle_xxx. Prior to ILOM 3.0.14, the Oracle Sun CIM classes should be referenced as Sun_xxxx and not Oracle_xxx as described in this guide. For further details about Oracle-supported CIM classes, see [“Oracle’s Sun-Supported CIM Classes” on page 153](#).

For a list of the supported DMTF profiles, Oracle-specific CIM classes, and the supported CIM indications in ILOM, see these sections:

- [“Supported DMTF SMASH Profiles and CIM Classes” on page 148](#)
- [“Supported CIM Indications” on page 150](#)

Supported DMTF SMASH Profiles and CIM Classes

As of ILOM 3.0.8, Oracle ILOM supports the following DMTF SMASH profiles and CIM classes.

Note – For viewing the published documentation on a supported DMTF profile, go to the DMTF Standards Publication site http://www.dmtf.org/standards/published_documents and look for the DSP# listed in the following table.

TABLE: Supported SMASH Profiles and CIM Classes

Supported DMTF Profiles	Oracle-Supported CIM Classes	Oracle-Derived Classes
Base Server (DSP1004)	<ul style="list-style-type: none"> • CIM_ComputerSystem • CIM_EnabledLogicalElementCapabilities • CIM_ElementCapabilities • CIM_ComputerSystemPackage • CIM_ElementConformsToProfile • CIM_SystemDevice • CIM_UseOfLog 	<ul style="list-style-type: none"> • Oracle_ComputerSystem • Oracle_EnabledLogicalElementCapabilities • Oracle_ElementCapabilities • Oracle_ComputerSystemPackage • Oracle_ElementConformsToProfile • Oracle_SystemDevice • Oracle_UseOfLog
Service Processor	<ul style="list-style-type: none"> • CIM_ComputerSystem • CIM_EnabledLogicalElementCapabilities • CIM_ElementCapabilities • CIM_SystemComponent 	<ul style="list-style-type: none"> • Oracle_ComputerSystem • Oracle_EnabledLogicalElementCapabilities • Oracle_ElementCapabilities • Oracle_SystemComponent
Physical Asset (DSP1011)	<ul style="list-style-type: none"> • CIM_Chip • CIM_PhysicalMemory • CIM_Chassis • CIM_PhysicalPackage • CIM_PhysicalAssetCapabilities • CIM_Container • CIM_Realizes • CIM_ComputerSystemPackage • CIM_ElementCapabilities 	<ul style="list-style-type: none"> • Oracle_Chip • Oracle_PhysicalMemory • Oracle_Chassis • Oracle_PhysicalPackage • Oracle_PhysicalAssetCapabilities • Oracle_Container • Oracle_Realizes • Oracle_ComputerSystemPackage • Oracle_ElementCapabilities
Sensors (DSP1009)	<ul style="list-style-type: none"> • CIM_Sensor • CIM_NumericSensor • CIM_AssociatedSensor • CIM_SystemDevice 	<ul style="list-style-type: none"> • Oracle_Sensor • Oracle_NumericSensor • Oracle_AssociatedSensor • Oracle_SystemDevice
CPU (DSP1022)	<ul style="list-style-type: none"> • CIM_Processor • CIM_Realizes • CIM_SystemDevice 	<ul style="list-style-type: none"> • Oracle_Processor • Oracle_Realizes • Oracle_SystemDevice
System Memory (DSP1026)	<ul style="list-style-type: none"> • CIM_Memory • CIM_Realizes • CIM_SystemDevice 	<ul style="list-style-type: none"> • Oracle_Memory • Oracle_Realizes • Oracle_SystemDevice

TABLE: Supported SMASH Profiles and CIM Classes *(Continued)*

Supported DMTF Profiles	Oracle-Supported CIM Classes	Oracle-Derived Classes
Indicator LED (DSP0835)	<ul style="list-style-type: none"> • CIM_SystemDevice • Not Applicable* • Not Applicable* <p>Note - *The CIM schema version 1.18.1 does not have the CIM_IndicatorLED and CIM_AssociatedIndicatorLED defined. The CIM_IndicatorLED and CIM_AssociatedIndicatorLED are required by the Indicator LED profile.</p>	<ul style="list-style-type: none"> • Oracle_SystemDevice • Oracle_IndicatorLED* • Oracle_AssociatedIndicatorLED* <p>Note - *Use the CIM_IndicatorLED and CIM_AssociatedIndicatorLED from the experimental schema for CIM schema version 2.18.1 and rename them Oracle_IndicatorLED and Oracle_AssociatedIndicatorLED.</p>
Record Log (DSP0810)	<ul style="list-style-type: none"> • CIM_RecordLog • CIM_LogEntry • CIM_LogManagesRecord • CIM_UseOfLog 	<ul style="list-style-type: none"> • Oracle_RecordLog • Oracle_LogEntry • Oracle_LogManagesRecord • Oracle_UseOfLog
Profile Registration (DSP1033)	<ul style="list-style-type: none"> • CIM_RegisteredProfile • CIM_ElementConformsToProfile • CIM_ReferenceProfile 	<ul style="list-style-type: none"> • Oracle_RegisteredProfile • Oracle_ElementConformsToProfile • Oracle_ReferenceProfile

Supported CIM Indications

As of ILOM 3.0.8, ILOM can generate CIM indications for the following conditions:

- Sensor crosses a threshold (CIM_ThresholdIndication).
- Hardware component changes operational state or health state (CIM_InstModification).
- Hardware component is inserted into the chassis (CIM_InstCreation).
- Hardware component is removed from the chassis (CIM_InstDeletion).

The following table identifies the CIM classes supported in Oracle ILOM for CIM indications.

TABLE: Oracle's Sun-Supported CIM Classes for Sensor Indications

Oracle's Sun-Supported CIM Classes for Sensor Indications	Oracle' Derived Classes for Sensor Indications
• CIM_InstCreation	• Oracle_InstCreation
• CIM_InstDeletion	• Oracle_InstDeletion
• CIM_InstModification	• Oracle_HWCompErrorOkIndication
• CIM_ThresholdIndication	• Oracle_ThresholdIndication

In addition, ILOM defines two static instances of `CIM_IndicationFilter`, in `/root/interop` namespace that a client can subscribe to in order to receive indication for when a threshold is crossed or for when a hardware component health state changes.

The following table identifies the key properties and ILOM values supported for these conditions.

TABLE: Key Properties and Values for Static `CIM_IndicationFilter` Instances

Key Property	ILOM Value
Subscription for sensor crossing threshold	
• <code>CreationClassName</code>	• <code>CIM_IndicationFilter</code>
• <code>Name</code>	<ul style="list-style-type: none"> • <code>ORCL:ILOM:SensorCrossingThresholdFilter</code> (as of ILOM 3.0.14) • <code>JAVA:ILOM:SensorCrossingThresholdFilter</code> (prior to ILOM 3.0.14)
• <code>SystemCreationClassName</code>	• <code>CIM_ComputerSystem</code>
• <code>SystemName</code>	• <code>localhost</code>
Subscription for hardware component changes health state	
• <code>CreationClassName</code>	• <code>CIM_IndicationFilter</code>
• <code>Name</code>	<ul style="list-style-type: none"> • <code>ORCL:ILOM:HWComponentErrorFilter</code> (as of ILOM 3.0.14) • <code>JAVA:ILOM:HWComponentErrorFilter</code> (prior to ILOM 3.0.14)
• <code>SystemCreationClassName</code>	• <code>CIM_ComputerSystem</code>
• <code>SystemName</code>	• <code>localhost</code>

Oracle's Sun-Supported CIM Classes

-
- | | |
|---|--|
| • “Document Conventions For Oracle’s Sun-Supported CIM Classes” on page 154 | • “Oracle_NumericSensor” on page 199 |
| • “Oracle_AssociatedIndicatorLED” on page 154 | • “Oracle_PhysicalAssetCapabilities” on page 207 |
| • “Oracle_AssociatedSensor” on page 156 | • “Oracle_PhysicalComponent” on page 209 |
| • “Oracle_Chassis” on page 157 | • “Oracle_PhysicalElementCapabilities” on page 215 |
| • “Oracle_ComputerSystem” on page 163 | • “Oracle_PhysicalMemory” on page 216 |
| • “Oracle_ComputerSystemPackage” on page 170 | • “Oracle_PhysicalPackage” on page 220 |
| • “Oracle_Container” on page 171 | • “Oracle_Processor” on page 227 |
| • “Oracle_ElementCapabilities” on page 172 | • “Oracle_ProcessorChip” on page 233 |
| • “Oracle_ElementConformsToProfile” on page 173 | • “Oracle_Realizes” on page 237 |
| • “Oracle_EnabledLogicalElementCapabilities” on page 174 | • “Oracle_RegisteredProfile” on page 238 |
| • “Oracle_HWCompErrorOkIndication” on page 177 | • “Oracle_RecordLog” on page 241 |
| • “Oracle_IndicatorLED” on page 178 | • “Oracle_ReferencedProfile” on page 246 |
| • “Oracle_InstCreation” on page 187 | • “Oracle_Sensor” on page 247 |
| • “Oracle_InstDeletion” on page 188 | • “Oracle_SpSystemComponent” on page 253 |
| • “Oracle_LogEntry” on page 189 | • “Oracle_SystemDevice” on page 254 |
| • “Oracle_LogManagesRecord” on page 193 | • “Oracle_ThresholdIndication” on page 255 |
| • “Oracle_Memory” on page 194 | • “Oracle_UseOfLog” on page 261 |
-

Related Information

- “Server Management Using WS-Management and CIM” on page 141

Document Conventions For Oracle's Sun-Supported CIM Classes

The following document conventions apply to Oracle Sun CIM classes presented in this section:

- Each class table within this section describes only the properties supported by ILOM. For all possible properties of a class, see the DMTF CIM Schema 2.18.1 at: http://www.dmtf.org/standards/cim/cim_schema_v2181
- An Oracle Sun-specific property (added by Oracle Sun-derived classes) will have the word *Sun-specific*.
- Key-property rows are shown first in each class table, in alpha-numerical ascending order.
- Non-key-property rows are after key-property rows, in alpha-numerical ascending order.
- The term *controller* refers to the hardware entity on which management software resides, for example, the service processor (SP) or chassis monitoring module (CMM). The term *controllee* refers to the hardware entity that is controlled by the controller, for example, the host system (SYS) or the chassis (CH).
- As of ILOM 3.0.14, the Oracle Sun CIM classes have been renamed from Sun_xxx to Oracle_xxx. Prior to ILOM 3.0.14, the Oracle Sun CIM classes should be referenced as Sun_xxxx and *not* Oracle_xxx as described in this guide.

Oracle_AssociatedIndicatorLED

Description:	The Oracle_AssociatedIndicatorLED class associates an LED to a physical element.
---------------------	--

Inheritance:	CIM_Dependency
Properties:	For a description of the supported properties for the Oracle_AssociatedIndicatorLED class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Indicator LED

Note – The Indicator LED profile specifies the CIM_AssociatedIndicatorLED class. However, the CIM_AssociatedIndicatorLED class does not exist in version 2.1.8.1 of the CIM Schema. Therefore, Oracle uses the CIM_AssociatedIndicatorLED class that is identified in the Experimental CIM Schema 2.18.1 and renamed it to Oracle_AssociatedIndicatorLED.

TABLE: Properties for Oracle_AssociatedSensor

Property	Data Type	Description	ILOM Value
Antecedent	CIM_ManagedSystem REF	The Antecedent property is a mandatory <i>key</i> property. Indicates the ManagedSystemElement that has an associated LED.	Object path to an instance of CIM_ManagedSystemElement.
Dependent	Oracle_IndicatorLED REF	The -Dependent property is a mandatory <i>key</i> property. Represents the indicator LED of the managed element.	Object path to an instance of Oracle_IndicatorLED.

Oracle_AssociatedSensor

Description:	The Oracle_AssociatedSensor class associates a sensor to the physical element.
Inheritance:	CIM_AssociatedSensor
Properties:	<p>For a description of the supported properties for the Oracle_AssociatedSensor class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Sensor

TABLE: Properties for Oracle_AssociatedSensor

Property	Data Type	Description	ILOM Value
Antecedent	CIM_Sensor REF	<p>The Antecedent property is a mandatory key property.</p> <p>Represents the sensor for the managed element.</p>	Object path to an instance of CIM_Sensor.
Dependent	CIM_PhysicalElement REF	<p>The Dependent property is a mandatory key property.</p> <p>The ManagedSystemElement for which information is measured by the sensor.</p>	Object path to an instance of the CIM_PhysicalElement that the sensor belongs.

Oracle_Chassis

Description:	The <code>Oracle_Chassis</code> class represents the physical elements that enclose other elements.
---------------------	---

Inheritance:	<code>CIM_Chassis</code>
---------------------	--------------------------

Properties:	<p>For a description of the supported properties for the <code>Oracle_Chassis</code> class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
--------------------	---

Profile:	Physical Asset
-----------------	----------------

TABLE: Properties for Oracle_Chassis

Property	Data Type	Description	ILOM Value
CreationClassName	string	<p>The CreationClassName property is a mandatory key property</p> <p>CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.</p>	Set to Oracle_Chassis
Tag	string	<p>The Tag property is a mandatory key property.</p> <p>The Tag property is an arbitrary string that uniquely identifies the physical element and serves as the key of the element.</p> <p>The Tag property can contain information such as asset tag or serial number data.</p> <p>The <i>key</i> for PhysicalElement is placed very high in the object hierarchy in order to independently identify the hardware or entity, regardless of physical placement in or on cabinets, adapters, and so on.</p> <p>For example, a hot-swappable or removable component can be taken from its containing (scoping) Package and be temporarily unused. The object still continues to exist and can be inserted into a different scoping container. Therefore, the <i>key</i> for PhysicalElement is an arbitrary string and is defined independently of any placement or location-oriented hierarchy.</p>	Set to component NAC name
CanBeFRUed	boolean	<p>The CanBeFRUed property is a boolean that indicates whether this PhysicalElement can be FRUed (TRUE) or not (FALSE).</p>	Will be set to TRUE or FALSE depending on whether the component is considered to be a FRU by the platform.

TABLE: Properties for Oracle_Chassis (Continued)

Property	Data Type	Description	ILOM Value
ChassisPackageType	uint16[]	<p>The ChassisPackageType property indicates the physical form factor for the type of chassis.</p> <p>This property may have a value when the PackageType property contains the value 3 Chassis Frame. A value of 28 Blade Enclosure indicates the Chassis is designed to contain one or more PhysicalPackage(s) of PackageType 16 "Blade" or PackageType 17 "Blade Expansion".</p> <p>Definition type values include any of the following:</p> <p>{Unknown, Other, SMBIOS Reserved, Desktop, Low Profile Desktop, Pizza Box, Mini Tower, Tower, Portable, LapTop, Notebook, Hand Held, Docking Station, All in One, Sub Notebook, Space-Saving, Lunch Box, Main System Chassis, Expansion Chassis, SubChassis, Bus Expansion Chassis, Peripheral Chassis, Storage Chassis, SMBIOS Reserved, Sealed-Case PC, SMBIOS Reserved, CompactPCI, AdvancedTCA, Blade Enclosure, DMTF Reserved, Vendor Reserved}</p> <p>Values for the definition types are:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, ..., 0x8000..0xFFFF}</p>	Will be set to 17 (Main System Chassis)
ChassisTypeDescription	string	The ChassisTypeDescription is a string providing more information about the ChassisPackageType.	Will have appropriate description.
Description	string	Textual description of the object.	Will have appropriate description.

TABLE: Properties for Oracle_Chassis (Continued)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>The ElementName property is a user-friendly name.</p> <p>This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Set to component NAC name.

TABLE: Properties for Oracle_Chassis (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Will have appropriate value depending on whether the component is in error state or not.
Manufacturer	string	<p>The Manufacturer property is the name of the organization responsible for producing the PhysicalElement.</p> <p>This organization might be the entity from whom the element is purchased, but this is not necessarily true. The latter information is contained in the vendor property of CIM_Product.</p>	Will have appropriate value if the chassis is considered a FRU by the platform.

TABLE: Properties for Oracle_Chassis (Continued)

Property	Data Type	Description	ILOM Value
Model	string	The Model property is the name by which the PhysicalElement is generally known.	Will have appropriate value if the chassis is considered a FRU by the platform.
OperationalStatus	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element. Various operational statuses are defined. Many of the enumeration's values are self-explanatory. Enumeration definitions can include any of the following:</p> <p>{Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions are as follows: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	OperationalStatus[0] will have appropriate value depending on whether the component is in error state or not.
PartNumber	string	Part number assigned by the organization that is responsible for producing or manufacturing the PhysicalElement	Will have appropriate value if the chassis is considered a FRU by the platform.
SKU	string	The SKU property is the stock-keeping unit number for this PhysicalElement.	Will have appropriate value if the chassis is considered a FRU by the platform.
SerialNumber	string	The SerialNumber property is a manufacturer-allocated number used to identify the physical element.	Will have appropriate value if the chassis is considered a FRU by the platform.
StatusDescriptions	string	<p>That StatusDescriptions property describes the various OperationalStatus array values.</p> <p>For example, if -Stopping is the value assigned to OperationalStatus, then this property may contain an explanation as to why an object is being stopped.</p> <p>Note that entries in this array are correlated with those at the same array index in OperationalStatus.</p>	StatusDescriptions[0] will have appropriate description on the reason for the value of OperationalStatus[0]

Oracle_ComputerSystem

Description:	The Oracle_ComputerSystem class represents a special collection of Sun system managed elements. This collection provides computer capabilities and serves as an aggregation point to associate one or more of the following elements: file system, operating system, processor and memory (volatile and non-volatile storage).
Inheritance:	CIM_ComputerSystem
Properties	<p>For a description of the supported properties for the Oracle_ComputerSystem class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profiles	<ul style="list-style-type: none">• Base Server• Service Processor

TABLE: Attributes for Oracle_ComputerSystem

Property	Data Type	Description	ILOM Value
CreationClassName	string	<p>The CreationClassName property is a mandatory key property.</p> <p>CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.</p>	Set value to: Oracle_ComputerSystem.
Name	string	<p>The Name attribute is a mandatory key CIM property.</p> <p>The inherited Name serves as the key of a system instance in an enterprise environment.</p>	Implementation-dependent value representing unique ID of the ComputerSystem.
Dedicated[]	string	<p>The Dedicated[] property enumerates the purposes to which the ComputerSystem is dedicated, if any, and what functionality is provided.</p> <p>Functionality definitions can include any of the following:</p> <p>{Not Dedicated, Unknown, Other, Storage, Router, Switch, Layer 3 Switch, Central Office Switch, Hub, Access Server, Firewall, Print, I/O, Web Caching, Management, Block Server, File Server, Mobile User Device, Repeater, Bridge/Extender, Gateway, Storage Virtualizer, Media Library, ExtenderNode, NAS Head, Self-contained NAS, UPS, IP Phone, Management Controller, Chassis Manager, Host-based RAID controller, Storage Device Enclosure, Desktop, Laptop, Virtual Tape Library, Virtual Library System, DMTF Reserved, Vendor Reserved}</p> <p>Values for these functionality definitions are as follows:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36..32567, 32568..65535}</p>	<p>For ComputerSystem instance representing the controller, the Dedicated[0] value will be set to -28 (Management Controller).</p> <p>For ComputerSystem instance representing the controllee, Dedicated[0] to 0 (Not Dedicated).</p>

TABLE: Attributes for Oracle_ComputerSystem (Continued)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>The ElementName property is a user-friendly name.</p> <p>This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	<p>For ComputerSystem instance representing the controller, the ElementName will be set to the controller or host name.</p> <p>For ComputerSystem instance representing the controllee, the ElementName will be set to the host product name.</p>
EnabledDefault	string	<p>The EnabledDefault property is an enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is Enabled (value=2).</p> <p>Element definitions include any of the following: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p> <p>Values for the element definitions are as follows: {2, 3, 5, 6, 7, 9, .., 32768..65535}</p>	<p>EnabledDefault will be set to default value 2 (Enabled).</p>

TABLE: Attributes for Oracle_ComputerSystem (Continued)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>EnabledState is an integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, Shutting Down (value=4) and Starting (value=10) are transient states between enabled and disabled. The following text briefly summarizes the various enabled and disabled states:</p> <ul style="list-style-type: none">• Enabled (2) indicates that the element is or could be executing commands, will process any queued commands, and queues new requests.• Disabled (3) indicates that the element will not execute commands and will drop any new requests.• Shutting Down (4) indicates that the element is in the process of going to a disabled state.• Not Applicable (5) indicates the element does not support being enabled or disabled.• Enabled but Offline (6) indicates that the element might be completing commands, and will drop any new requests.• Test (7) indicates that the element is in a test state.• Deferred (8) indicates that the element might be completing commands, but will queue any new requests.• Quiesce (9) indicates that the element is enabled but in a restricted mode.• Starting (10) indicates that the element is in the process of going to an enabled state. New requests are queued. <p>The following values apply: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11..32767, 32768..65535}</p> <p>Value definitions include: {Unknown, Other, Enabled, Disabled, Shutting Down, Not Applicable, Enabled but Offline, In Test, Deferred, Quiesce, Starting, DMTF Reserved, Vendor Reserved}</p>	<p>For ComputerSystem instance representing the controller, the EnabledState value will be set to 2 (Enabled).</p> <p>For ComputerSystem instance representing the controllee, the EnabledState value will be set to appropriate value depending on the power state of the controllee.</p>

TABLE: Attributes for Oracle_ComputerSystem (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	<p>For ComputerSystem instance representing the controller EnabledState will be set to 5 (OK).</p> <p>For ComputerSystem instance representing the controllee HealthState will be set to appropriate value depending on the value of OperationalStatus property.</p>
IdentifyingDescriptions	string	<p>The IdentifyingDescriptions property is an array of free-form strings providing explanations and details behind the entries in the OtherIdentifyingInfo array.</p> <p>Note - Each entry of this array is related to the entry in OtherIdentifyingInfo that is located at the same index.</p>	<p>For ComputerSystem instance representing the controller, the IdentifyingDescriptions will not be set.</p> <p>For ComputerSystem instance representing the controllee, the IdentifyingDescriptions will be set to the following value: "-CIM:Model:SerialNumber".</p>

TABLE: Attributes for Oracle_ComputerSystem (Continued)

Property	Data Type	Description	ILOM Value
OperationalStatus	uint16[]	<p>The OperationalStatus indicates the current statuses of the element. Various operational statuses are defined. Many of the enumeration's values are self-explanatory. However, a few are not and are described here in more detail in the CIM_ComputerSystem.mof described in the DMTF CIM schema v2.18.1.</p> <p>Element definitions include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for the above definitions are: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	<p>For ComputerSystem instance representing the controller, the OperationalStatus[0] will be set to 2 (OK).</p> <p>For ComputerSystem instance representing the controllee, the OperationalStatus[0] will be set to appropriate value depending on the power status (starting, stopping) or whether the host has incurred an error or is unknown.</p>
OtherEnabledState	string	<p>The OtherEnabledState property is a string that describes the enabled or disabled state of the element when the EnabledState property is set to 1 (Other). This property must be set to null when EnabledState is any value other than 1.</p>	Will be set to empty string.

TABLE: Attributes for Oracle_ComputerSystem (Continued)

Property	Data Type	Description	ILOM Value
OtherIdentifyingInfo	String[]	The OtherIdentifyingInfo property captures additional data, beyond system name information, that could be used to identify a ComputerSystem. One example would be to hold the Fibre Channel World-Wide Name (WWN) of a node. Note that if only the Fibre Channel name is available and is unique (able to be used as the system key), then this property would be NULL and the WWN would become the system key, its data placed in the Name property.	For ComputerSystem instance representing the controller, the OtherIdentifyingInfo will not be set. For ComputerSystem instance representing the controllee, the OtherIdentifyingInfo[0] will be set to the following value: <product-name>:<Serial Number> For more details, refer to the DMTF Base Server Profile.
RequestedState	uint16[]	The RequestedState property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled or disabled states. Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved} Values for these definitions are as follows: {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ..., 32768..65535} Note - When EnabledState is set to 5 (Not Applicable), then this property has no meaning.	Will set to -Not -Applicable if there has not been any invocation of RequestStateChange() method. Will be set to the correct value of the incoming argument of RequestStateChange().
RequestStateChange()	UInt32	Method for client to request state change. The following state change operations are as follows: {2, 3, 4, 6, 7, 8, 9, 10, 11, ..., 32768..65535} Definitions for the above values are as follows: {Enabled, Disabled, Shut Down, Offline, Test, Defer, Quiesce, eboot, Reset, DMTF Reserved, Vendor Reserved"}	For ComputerSystem instance representing the controller, will support 11 (Reset). For ComputerSystem instance representing the controllee, will support 2 (Enabled), 3 (Disabled), 4 (Shut Down). This operation is supported only if the user has Admin role.

Oracle_ComputerSystemPackage

Description:	The Oracle_ComputerSystemPackage class is used to associate the instance of Oracle_ComputerSystem representing the controllee to the physical package chassis, Oracle_Chassis, which realizes the Oracle_ComputerSystem.
Inheritance:	CIM_ComputerSystemPackage
Properties:	For a description of the supported properties for the Oracle_ComputerSystemPackage class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	<ul style="list-style-type: none">• Physical Asset• Base Server

TABLE: Properties for Oracle_ComputerSystemPackage

Property	Data Type	Description	ILOM Value
Antecedent	Oracle_Chassis REF	The Antecedent property is a mandatory key property. The chassis that realizes a Oracle_ComputerSystem.	Object path to an instance of Oracle_Chassis.
Dependent	Oracle_ComputerSystem REF	The Dependent property is a mandatory key property. Represents the Oracle_ComputerSystem.	Object path to the instance of Oracle_ComputerSystem representing the controllee.

Oracle_Container

Description:	Oracle_Container is used to associate a physical package (CIM_PhysicalPackage) and a physical element (CIM_PhysicalElement) contained in the physical package.
Inheritance:	CIM_Container
Properties:	For a description of the supported properties for the Oracle_Container class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Physical Asset

TABLE: Properties for Oracle_Container

Property	Data Type	Description	ILOM Value
GroupComponent	CIM_PhysicalPackage REF	The GroupComponent property is a mandatory key property. The PhysicalPackage that contains other physical elements, including other packages.	Object path to an instance of CIM_PhysicalPackage.
PartComponent	CIM_PhysicalElement REF	The PartComponent property is a mandatory key property. The PhysicalElement that is contained in the package.	Object path to an instance of CIM_PhysicalElement.

Oracle_ElementCapabilities

Description:	The Oracle_ElementCapabilities class is used to associate an instance of ManagedElements and its capabilities.
Inheritance:	CIM_ElementCapabilities
Properties:	For a description of the supported properties for the Oracle_ElementCapabilities class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Base Server

TABLE: Properties for Oracle_ElementCapabilities

Property	Data Type	Description	ILOM Value
Capabilities	CIM_Capabilities REF	The Capabilities property is a mandatory key property. The Capabilities object that is associated with the element.	Object path to an instance of Oracle_EnabledLogicalElementCapabilities.
ManagedElement	CIM_ManagedElement REF	The ManagedElement property is a mandatory key property. Identifies the managed element.	Object path to an instance of Oracle_ComputerSystem.

Oracle_ElementConformsToProfile

Description:	Oracle_ElementConformsToProfile associates the instance of Oracle_ComputerSystem representing the controllee to the instance of Oracle_RegisteredProfile representing the Base Server Profile.
Inheritance:	CIM_ElementConformsToProfile
Properties:	For a description of the supported properties for the Oracle_ElementConformsToProfile class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Profile Registration Base Server

TABLE: Properties for Oracle_ElementConformsToProfile

Property	Data Type	Description	ILOM Value
ElementConformsToProfile	Oracle_RegisteredProfile REF	The ElementconformsToProfile property is a mandatory <i>key</i> property. The RegisteredProfile to which the ManagedElement conforms.	Object path to the instance of Oracle_RegisteredProfile .
ManagedElement	Oracle_ComputerSystem REF	The ManagedElement property is a mandatory <i>key</i> property. The Oracle_ComputerSystem.	Object path to the instance of Oracle_ComputerSystem representing the controllee.

Oracle_EnabledLogicalElementCapabilities

Description:	EnabledLogicalElementCapabilities describes the capabilities supported for changing the state of the associated EnabledLogicalElement.
Inheritance:	CIM_EnabledLogicalElementCapabilities
Properties:	<p>For a description of the supported properties for the Oracle_EnabledLogicalElementCapabilities class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Base Server

TABLE: Properties for Oracle_EnabledLogicalElementCapabilities

Property	Data Type	Description	ILOM Value
Instance ID	string	<p>The InstanceID property is a mandatory key property. Within the scope of the instantiating Namespace, the InstanceID property uniquely identifies an instance of this class. The value of InstanceID should be constructed using the following preferred algorithm:</p> <pre><OrgID>:<LocalID></pre> <p>Where:</p> <ul style="list-style-type: none"> • <OrgID> and <LocalID> are separated by a colon (:) • <OrgID> must include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating or defining InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority. (This is similar to the <Schema Name>_<Class Name> structure of schema class names.) • <OrgID> must not contain a colon (:). The first colon to appear in InstanceID must appear between <OrgID> and <LocalID>. • <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements. • For DMTF defined instances, the <OrgID> must be set to CIM. <p>If this preferred algorithm is not used, the defining entity must ensure that the resultant InstanceID is not re-used across any instance IDs produced by this or other providers for this instance's NameSpace.</p>	Represents the unique ID of the EnabledLogicalElementCapabilities.
Description	string	Textual description of the object.	Appropriate descriptions.

TABLE: Properties for Oracle_EnabledLogicalElementCapabilities (*Continued*)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>The ElementName property is a user-friendly name.</p> <p>This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note that the Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Appropriate value.
ElementNameEditsSupported	boolean	The boolean indicates whether the ElementName can be modified.	Set to False.
RequestedStatesSupported	uint16[]	<p>Indicates the possible states that can be requested when using the method RequestStateChange on the EnabledLogicalElement. The following values apply: {2, 3, 4, 6, 7, 8, 9, 10, 11}</p> <p>Definitions for these values are as follows: {Enabled, Disabled, Shut Down, Offline, Test, Defer, Quiesce, Reboot, Reset}</p>	<p>For the EnabledLogicalElementCapabilities instance representing the controller, RequestedStatesSupported[0] will be set to 11 (Reset).</p> <p>For the EnabledLogicalElementCapabilities instance representing the controllee, RequestedStatesSupported[] will be set to 2 (Enabled), 3 (Disabled), or 4 (Shut Down).</p>

Oracle_HWCompErrorOkIndication

Description:	When a client creates an indication subscription in which the filter indicates that it looks for <code>CIM_InstModification</code> in which the modified object is a <code>PhysicalElement</code> (that is the query statement is <code>SourceInstance ISA CIM_PhysicalElement</code>), and it looks for changes in <code>SourceInstance.OperationalStatus</code> or <code>SourceInstance.HealthState</code> then ILOM CIM-subsystem will generate <code>Oracle_HWCompErrorOkIndication</code> indication when a hardware component changes from good to bad, or vice versa.
Inheritance:	<code>CIM_InstModification</code>
Properties:	For a description of the supported properties for the <code>Oracle_HWCompErrorOkIndication</code> class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	None

TABLE: Properties for Oracle_HWCompErrorOkIndication

Property	Data Type	Description	ILOM Value
<code>PreviousInstance</code>	string	A copy of the previous instance whose change generated the indication. <code>PreviousInstance</code> contains older values of an instance's properties (as compared to <code>SourceInstance</code>), selected by the <code>IndicationFilter</code> 's query.	String representation of the previous instance of <code>CIM_PhysicalElement</code> that is affected.
<code>SensorObjectPath</code> (SUN-specific)	string	Object path of the sensor that causes the hardware component to change operational state.	Appropriate value.

TABLE: Properties for Oracle_HWCompErrorOkIndication (Continued)

Property	Data Type	Description	ILOM Value
SourceInstance	string	A copy of the instance that changed to generate the Indication. SourceInstance contains the current values of the properties selected by the indication filter's query. In the case of CIM_InstDeletion, the property values are copied before the instance is deleted.	String representation of the instance of CIM_PhysicalElement that is affected.
SourceInstanceHost	string	The host name or IP address of the SourceInstance.	Will have the value Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem representing the controllee.
SourceInstanceModelPath	string	The model path of the SourceInstance. The following format <i>must</i> be used to encode the model path: <NamespacePath>:<ClassName>.<Property1>=<Value1>, <Property2>=<Value2>, ..	String representation of the object path of the SourceInstance.



Oracle_IndicatorLED

Description:	The Oracle_IndicatorLED class models the logical aspects of an indicator LED.
Inheritance:	CIM_IndicatorLED
Properties:	For a description of the supported properties for the Oracle_IndicatorLED class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Indicator LED

Note – The Indicator LED profile specifies the `CIM_IndicatorLED` class. However, the `CIM_IndicatorLED` class does not exist in version 2.18.1 of the CIM Schema. Therefore, Oracle uses the `CIM_IndicatorLED` class that is identified in the Experimental CIM Schema version 2.18.1 and re-named it to `Oracle_IndicatorLED`.

TABLE: Properties for `Oracle_IndicatorLED`

Property	Data Type	Description	ILOM Value
<code>CreationClassName</code>	string	The <code>CreationClassName</code> property is a mandatory key property. <code>CreationClassName</code> indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to <code>Oracle_IndicatorLED</code> .
<code>DeviceID</code>	string	The <code>DeviceID</code> property is a mandatory key property. An address or other identifying information used to uniquely name the <code>LogicalDevice</code> .	Set to the NAC name of the of the LED.
<code>SystemCreationClassName</code>	string	The <code>SystemCreationClassName</code> property is a mandatory key property. The <code>SystemCreationClassName</code> of the scoping system.	Set to <code>Oracle_ComputerSystem</code> .
<code>SystemName</code>	string	The system name of the scoping system.	Will be set to <code>Oracle_ComputerSystem.Name</code> of the instance of <code>Oracle_ComputerSystem</code> that represents the controllee.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
ActivationState	uint16[]	<p>Indicates the current activity of an LED. An LED can exhibit behaviors that vary greatly in complexity. If the behavior of the LED is simple or a detailed depiction of the behavior is unnecessary to convey to client applications, values other than 5 (ControlPattern) can be used to indicate the behavior. If the behavior is complex and detailed information about the behavior is meaningful to a client, the value 5 (ControlPattern) can be used to indicate the ControlPattern property that describes the behavior. 2 (Lit) shall indicate that the LED is continuously illuminated without variation in color or intensity. 3 (Blinking) shall indicate that the LED is alternating between illuminated and not illuminated in a regular pattern without variation in color or intensity. The pattern is not indicated. 4 (Off) shall indicate that the LED is not illuminated. 5 (ControlPattern) shall indicate that the LED is exhibiting behavior that is described using the ControlPattern property.</p> <p>The following values apply: {2, 3, 4, 5, ..., 32768..65535}</p> <p>Definitions for these values are as follows: {Lit, Blinking, Off, Control Pattern, DMTF Reserved, Vendor Reserved}</p>	Appropriate value.
Color	uint16[]	<p>Indicates the current color of the LED. If the value of the ActivationState property is 4 (Off), this property will indicate the color of the LED the last time it was lit, or it will have the value 2 (Not Applicable).</p>	Appropriate value.
ControlMode	uint16[]	<p>Indicates the current control mode for the LED. 2 (Automatic) shall indicate that the state of the LED is being controlled by the management infrastructure. 3 (Manual) shall indicate that the state of the LED is being controlled by a management client. 4 (Test) shall indicate that the LED is in a test mode.</p> <p>The following values apply: {2, 3, 4, ..., 32768..65535}</p> <p>Definitions for these values are as follows: {Automatic, Manual, Test, DMTF Reserved, Vendor Reserved}</p>	Appropriate value.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
ControlPattern	string	<p>An LED can exhibit a range of behavior from very simple (for example, solid on) to very complicated (for example, a series of blinks of alternating color and duration). ControlPattern specifies the vendor or standard behavior exhibited by the LED if it cannot be described using one of the standard behaviors listed for the ActivationState property. If ActivationState has the value 5 (ControlPattern), the ControlPattern property shall not be NULL. The value of ControlPattern should be constructed using the following preferred algorithm:</p> <pre><OrgID>::<Pattern></pre> <p>where:</p> <ul style="list-style-type: none"> • <OrgID> and <Pattern> are separated by two colons (::) • <OrgID> includes a copyrighted, trademarked, or otherwise unique name that is owned by the business entity that is creating or defining the ControlPattern or that is a registered ID assigned to the business entity by a recognized global authority • If the definition of the value is specified by the DMTF, the value of <OrgID> must be DMTF. • <Pattern> is chosen by the business entity and should not be reused to identify different underlying (real-world) behaviors. If the behavior specified for the LED adheres to a standard or proprietary specification, <Pattern> should be a uniquely assigned value identifying the behavior. If the behavior for the LED is described using a standard or proprietary grammar, <Pattern> should be prefixed with a uniquely assigned identifier for the grammar. 	Appropriate value.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>Specifies an identifier for the LED. The value of ElementName should be constructed using the following preferred algorithm:</p> <p><OrgID> : <LocalID></p> <p>where:</p> <ul style="list-style-type: none">• <OrgID> and <LocalID> are separated by two colons (:)• <OrgID> includes a copyrighted, trademarked, or otherwise unique name that is owned by the business entity that is creating or defining the ControlPattern or that is a registered ID assigned to the business entity by a recognized global authority.• <LocalID> is chosen by the business entity and should not be reused to identify different underlying (real-world) elements.	Set to the NAC name of the LED.
EnabledDefault	uint16[]	<p>An enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is Enabled (value= 2).</p> <p>Valid values are as follows:</p> <p>{2, 3, 5, 6, 7, 9, ..., 32768..65535}</p> <p>Definitions for the valid values are:</p> <p>{Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p>	Set to default value 2 (Enabled).

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, -Shutting -Down (value=4) and -Starting (value=10) are transient states between enabled and disabled. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) • 1 (Other) • 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests. • 3 (Disabled) - The element will not execute commands and will drop any new requests • 4 (Shutting Down) - The element is in the process of going to a disabled state. • 5 (Not Applicable) - The element does not support being enabled or disabled. • 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests. • 7 (Test) - The element is in a test state. • 8 (Deferred) - The element might be completing commands, but will queue any new requests. • 9 (Quiesce) - The element is enabled but in a restricted mode. • 10 (Starting) - The element is in the process of going to an enabled state. New requests are queued. • 11..32767 (DMTF Reserved) • 32768..65539 (Vendor Reserved) 	Appropriate value.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents.</p> <p>The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown) - The implementation cannot report on <code>HealthState</code> at this time.• 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error.• 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors.• 15 (Minor Failure) - All functionality is available but some might be degraded.• 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working.• 25 (Critical Failure) - The element is non-functional and recovery might not be possible.• 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional <code>HealthStates</code> in the future.</p>	Appropriate value.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
IndicatedConditions	uint16[]	<p>The condition indicated by the LED.</p> <p>The following values apply:</p> <ul style="list-style-type: none">• 2 (Not Applicable) - The LED is currently not assigned an interpretation.• 3 (Location) - The LED is used to indicate that the location of associated managed elements.• 4 (Attention) - The LED is used to indicate that the associated managed elements requires the attention of service personnel.• 5 (Activity) -The LED is used to indicate that activity is occurring for the associated managed elements. The type of activity indicated is specific to the associated managed elements.• 6 (Powered On) - The LED is used to indicate if the associated managed elements are receiving power.• 7 (Fault) - The LED is used to indicate if the associated managed elements are in a fault, error, or otherwise degraded state.	Appropriate value.

TABLE: Properties for Oracle_IndicatorLED (*Continued*)

Property	Data Type	Description	ILOM Value
OperationalStatus	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration values can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>A list of valid values for the enumeration values include: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	Appropriate value.
OtherIndicatedConditionDescription	string	<p>This property will have a value if IndicatedCondition contains the value 1 (Other).</p>	<p>Will have appropriate value if IndicatedCondition contains the value 1 (Other).</p>
RequestedState	uint16[]	<p>The RequestedState property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for the above definitions include: {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, .., 32768..65535}</p> <p>Note - When EnabledState is set to 5 (Not Applicable), then this property has no meaning.</p>	Set to 12 (Not Applicable).

Oracle_InstCreation

Description:	When a client creates an indication subscription in which the filter indicates that it looks for CIM_InstCreation and SourceInstance is a PhysicalElement (for example, the query statement contains SourceInstance ISA CIM_PhysicalElement) then the Oracle ILOM CIM subsystem will generate an Oracle_InstCreation indication when it detects that a hardware component is hot inserted into the chassis.
Inheritance:	CIM_InstCreation
Properties:	For a description of the supported properties for the Oracle_InstCreation, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	None

TABLE: Properties for Oracle_InstCreation

Property	Data Type	Description	ILOM Value
SourceInstance	string	A copy of the instance that changed to generate the indication. SourceInstance contains the current values of the properties selected by the indication filter's query. In the case of CIM_InstDeletion, the property values are copied before the instance is deleted.	String representation of the instance of CIM_PhysicalElement that is hot-inserted.
SourceInstanceHost	string	The host name or IP address of the SourceInstance.	Will have the value Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem representing the controllee.
SourceInstanceModelPath	string	The model path of the SourceInstance. The following format must be used to encode the model path: <NamespacePath>:<ClassName>.<Prop1>=<Value1>, <Prop2>=<Value2>, ...	String representation of the object path of the SourceInstance.

Oracle_InstDeletion

Description:	When a client creates an indication subscription in which the filter indicates that it looks for CIM_InstDeletion and SourceInstance is a PhysicalElement (for example, the query statement contains SourceInstance ISA CIM_PhysicalElement), then the Oracle ILOM CIM subsystem will generate an Oracle_InstDeletion indication when it detects that a hardware component is hot-removed from the chassis.
Inheritance:	CIM_InstDeletion
Properties:	For a description of the supported properties for the Oracle_InstDeletion class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	None

TABLE: Properties for Oracle_InstDeletion

Property	Data Type	Description	ILOM Value
SourceInstance	string	A copy of the instance that changed to generate the indication. SourceInstance contains the current values of the properties selected by the indication filter's query. In the case of CIM_InstDeletion, the property values are copied before the instance is deleted.	String representation of the instance of CIM_PhysicalElement that is hot-removed.
SourceInstance Host	string	The host name or IP address of the SourceInstance.	Will have the value Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem representing the controllee.
SourceInstance ModelPath	string	The model path of the SourceInstance. The following format must be used to encode the model path: <NamespacePath>:<ClassName>.<Prop1>=<Value1>, <Prop2>=\"<Value2>\", ...	String representation of the object path of the SourceInstance.

Oracle_LogEntry

Description:	Oracle_LogEntry is used to represent individual log records of IPMI SEL log.
Inheritance:	CIM_LogEntry
Properties:	<p>For a description of the supported properties for the Oracle_LogEntry class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Record Log

TABLE: Properties for Oracle_LogEntry

Property	Data Type	Description	ILOM Value
InstanceID	string	<p>The InstanceID property is a mandatory key property.</p> <p>Within the scope of the instantiating Namespace, InstanceID uniquely identifies an instance of this class. In order to ensure uniqueness within the Namespace, the value of InstanceID should be constructed using the following preferred algorithm:</p> <p><OrgID> : <LocalID></p> <p>where:</p> <ul style="list-style-type: none"> • <OrgID> and <LocalID> are separated by a colon (:) • <OrgID> You must include a copyrighted, trademarked, or otherwise the unique name that is owned by the business entity creating or defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority. (This is similar to the <Schema Name>_<Class Name> structure of Schema class names.) • <OrgID> must not contain a colon (:). When you use this algorithm, the first colon to appear in InstanceID must appear between <OrgID> and <LocalID>. • <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements. • For DMTF defined instances, the preferred algorithm must be used with the <OrgID> set to CIM. <p>If this preferred algorithm is not used, the defining entity must ensure that the resultant InstanceID is not re-used across any InstanceIDs produced by this or other providers for this instance's Namespace.</p>	Implementation dependent value representing unique ID.
CreationTimeStamp	datetime	A LogEntry can include a time stamp for the entry.	Appropriate value.
Description	string	Textual description of the object.	SEL event description.

TABLE: Properties for Oracle_LogEntry (Continued)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	SEL event record ID.
LogInstance ID	string	The string containing the log's InstanceID.	Implementation dependent value representing unique ID of the associated Oracle_RecordLog.
LogName	string	The string containing the log's Name . This property is available for backwards continuity with CIM_LogRecord.	Will have the value SEL Log.

TABLE: Properties for Oracle_LogEntry (*Continued*)

Property	Data Type	Description	ILOM Value
RecordData	string	A string containing LogRecord data. If the corresponding RecordFormat property is <empty>, or cannot be parsed according to the recommended format, RecordData should be interpreted as a free-form string. If the RecordFormat property contains parseable format information (as recommended in the RecordFormat Description qualifier), the RecordData string should be parsed in accordance with this format. In this case, RecordData should begin with the delimiter character, and this character should be used to separate substrings in the manner described. The RecordData string can then be parsed by the data consumer and appropriately typed.	Contents of the SEL event data.
RecordFormat	string	<p>A string describing the data structure of the information in the property, RecordData. If the RecordFormat string is <empty>, RecordData should be interpreted as a free-form string. To describe the data structure of RecordData, the RecordFormat string should be constructed as follows:</p> <ul style="list-style-type: none"> • The first character is a delimiter character and is used to parse the remainder of the string into sub-strings. • Each substring is separated by the delimiter character and should be in the form of a CIM property declaration (for example, data type and property name). This set of declarations can be used to interpret the similarly delimited RecordData property. <p>For example, using a * delimiter: RecordFormat = <i>"*string ThisDay*uint32 ThisYear*datetime SomeTime"</i> can be used to interpret: RecordData = <i>"*This is Friday*2002*20020807141000.000000-300"</i>.</p>	Will have the format used for interpreting the RecordData property.
RecordID	string	Provides a representation of log entry ordering or pointers and handles for log entries.	SEL event record ID.

Oracle_LogManagesRecord

Description:	Oracle_LogManagesRecord is used to associate the instance of Oracle_RecordLog representing the IPMI SEL log to an instance of the SEL log record.
Inheritance:	CIM_LogManagesRecord
Properties:	For a description of the supported properties for the Oracle_LogManagesRecord class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Record log

TABLE: Properties for Oracle_LogManagesRecord

Property	Data Type	Description	ILOM Value
Log	Oracle_RecordLog REF	The Log property is a mandatory key property. Indicates the Oracle_RecordLog.	Object path to the instance of Oracle_RecordLog representing the IPMI SEL log.
Record	Oracle_LogEntry REF	The Record property is a mandatory key property. Indicates the Oracle_LogEntry.	Object path to an instance of Oracle_LogEntry.

Oracle_Memory

Description:	Provides capabilities and management of memory-related LogicalDevices.
Inheritance:	CIM_Memory
Properties:	For a description of the supported properties for the Oracle_Memory class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	System Memory

TABLE: Properties for Oracle_Memory

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_Memory.
DeviceID	string	The DeviceID property is a mandatory <i>key</i> property. An address or other identifying information used to uniquely name the LogicalDevice.	Implementation dependent value representing unique ID.
SystemCreationClassName	string	The SystemCreationClassName property is a mandatory key property. Indicates the SystemCreationClassName of the scoping system.	Set to Oracle_ComputerSystem.
SystemName	string	The SystemName property is a mandatory key property. Indicates the SystemName of the scoping system.	Will be set to Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem that represents the controllee.

TABLE: Properties for Oracle_Memory (Continued)

Property	Data Type	Description	ILOM Value
Access	uint16[]	<p>The <code>Access</code> property describes whether the media is <i>readable</i> (value=1), <i>writable</i> (value=2), or both (value=3). <i>Unknown</i> (0) and <i>Write Once</i> (4) can also be defined.</p> <p>The following values apply: {0, 1, 2, 3, 4}</p> <p>Definitions for these values are: {Unknown, Readable, Writable, Read/Write Supported, Write Once}</p>	Set to 3 (Read/Write Supported).
BlockSize	uint16[]	<p>Size in bytes of the blocks that form this <code>StorageExtent</code>. If the block size is variable, then the maximum block size in bytes should be specified. If the block size is unknown or if a block concept is not valid (for example, for <code>AggregateExtents</code>, <code>Memory</code> or <code>LogicalDisks</code>), enter a 1.</p>	Set to appropriate value if memory size can be computed.
ElementName	string	<p>The <code>ElementName</code> property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The <code>Name</code> property of <code>ManagedSystemElement</code> is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where <code>Name</code> exists and is not a key (such as for instances of <code>LogicalDevice</code>), the same information can be present in both the <code>Name</code> and <code>ElementName</code> properties.</p>	Appropriate value.
EnabledDefault	uint16[]	<p>Enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is 2 (Enabled).</p> <p>The following values apply: {2, 3, 5, 6, 7, 9, ..., 32768..65535}</p> <p>Definitions for these values are: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p>	Set to default value 2 (Enabled).

TABLE: Properties for Oracle_Memory (*Continued*)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, -Shutting -Down (value=4) and -Starting (value=10) are transient states between enabled and disabled.</p> <p>The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown)• 1 (Other)• 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests.• 3 (Disabled) - The element will not execute commands and will drop any new requests.• 4 (Shutting Down) - The element is in the process of going to a disabled state.• 5 (Not Applicable) - The element does not support being enabled or disabled.• 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests.• 7 (Test) - The element is in a test state.• 8 (Deferred) - The element might be completing commands, but will queue any new requests.• 9 (Quiesce) - The element is enabled but in a restricted mode.• 10 (Starting) - The element is in the process of going to an enabled state. New requests are queued.• 11..32767 (DMTF Reserved)• 32768..65539 (Vendor Reserved)	Appropriate value.

TABLE: Properties for Oracle_Memory (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents.</p> <p>The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown) - The implementation cannot report on HealthState at this time.• 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error.• 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors.• 15 (Minor Failure) - All functionality is available but some might be degraded.• 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working.• 25 (Critical Failure) - The element is non-functional and recovery might not be possible.• 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Appropriate value.

TABLE: Properties for Oracle_Memory (Continued)

Property	Data Type	Description	ILOM Value
NumberOfBlocks	uint16[]	Total number of logically contiguous blocks, of size <code>BlockSize</code> , which form this extent. The total size of the extent can be calculated by multiplying <code>BlockSize</code> by <code>NumberOfBlocks</code> . If the <code>BlockSize</code> is 1, this property is the total size of the extent.	Will have appropriate value if memory size can be computed.
OperationalStatus	uint16[]	<p>The <code>OperationalStatus</code> property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration values can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Possible values for the enumeration values include: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	Appropriate value.
RequestedState	uint16[]	<p>The <code>RequestedState</code> property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by <code>EnabledState</code>.</p> <p>This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions include: {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, .., 32768..65535}</p> <p>Note - When <code>EnabledState</code> is set to 5 (Not Applicable), then this property has no meaning.</p>	Set to 12 (Not Applicable).

Oracle_NumericSensor

Description:	A numeric sensor that returns numeric readings and optionally supports thresholds settings.
Inheritance:	CIM_NumericSensor
Properties:	For a description of the supported properties for the Oracle_NumericSensor class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Sensor

TABLE: Properties for Oracle_NumericSensor

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_NumericSensor.
DeviceID	string	The DeviceID property is a mandatory key property. An address or other identifying information used to uniquely name the LogicalDevice.	Set to the NAC name of the sensor.
SystemCreationClassName	string	The SystemCreationClassName property is a mandatory key property. Indicates the CreationClassName for the scoping system.	Will be set to Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem that represents the controllee.

TABLE: Properties for Oracle_NumericSensor (Continued)

Property	Data Type	Description	ILOM Value
SystemName	string	The SystemName property is a mandatory key property. Indicates the SystemName of the scoping system.	Set to Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem that represents the controllee.
BaseUnits	uint16[]	<p>The base unit of the values returned by this sensor. All the values returned by this sensor are represented in the units obtained by BaseUnits * 10 raised to the power of the UnitModifier. For example, if BaseUnits is <i>Volts</i> and the UnitModifier is -6, then the units of the values returned are microvolts. However, if the RateUnits property is set to a value other than <i>None</i>, then the units are further qualified as rate units. In this example, if RateUnits is set to <i>Per Second</i>, then the values returned by the sensor are in microvolts/second. The units apply to all numeric properties of the sensor, unless explicitly overridden by the units qualifier.</p> <p>The following values apply:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66}</p> <p>Definitions of these values are:</p> <p>{Unknown, Other, Degrees C, Degrees F, Degrees K, Volts, Amps, Watts, Joules, Coulombs, VA, Nits, Lumens, Lux, Candelas, kPa, PSI, Newtons, CFM, RPM, Hertz, Seconds, Minutes, Hours, Days, Weeks, Mils, Inches, Feet, Cubic Inches, Cubic Feet, Meters, Cubic Centimeters, Cubic Meters, Liters, Fluid Ounces, Radians, Steradians, Revolutions, Cycles, Gravities, Ounces, Pounds, Foot-Pounds, Ounce-Inches, Gauss, Gilberts, Henries, Farads, Ohms, Siemens, Moles, Becquerels, PPM (parts/million), Decibels, DbA, DbC, Grays, Sieverts, Color Temperature Degrees K, Bits, Bytes, Words (data), DoubleWords, QuadWords, Percentage, Pascals}</p>	Appropriate value depending on sensor type.
CurrentReading	sint32	The current value indicated by the sensor.	Appropriate value.
CurrentState	string	The current state indicated by the sensor. This is always one of the PossibleStates.	Appropriate value representing current state of the sensor.

TABLE: Properties for Oracle_NumericSensor (Continued)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is <i>not a key</i> (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Set to the NAC name of the sensor.
EnabledDefault	uint16[]	<p>An enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is Enabled (value=2).</p> <p>The following values apply: {2, 3, 5, 6, 7, 9, ..., 32768..65535}</p> <p>Definitions of these values are: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p>	Set to default value 2 (Enabled).

TABLE: Properties for Oracle_NumericSensor (*Continued*)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, shutting down (value=4) and starting (value=10) are transient states between enabled and disabled. The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown)• 1 (Other)• 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests.• 3 (Disabled) - The element will not execute commands and will drop any new requests.• 4 (Shutting Down) - The element is in the process of going to a disabled state.• 5 (Not Applicable) - The element does not support being enabled or disabled.• 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests.• 7 (Test) - The element is in a test state.• 8 (Deferred) - The element might be completing commands, but will queue any new requests.• 9 (Quiesce) - The element is enabled but in a restricted mode.• 10 (Starting) - The element is in the process of going to an Enabled state. New requests are queued.• 11..32767 (DMTF Reserved)• 32768..65539 (Vendor Reserved)	Will have appropriate value depending on whether the sensor is enabled, disabled, or unknown.

TABLE: Properties for Oracle_NumericSensor (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Appropriate value.
LowerThresh oldCritical	sint32	The sensor's threshold values specify the ranges (min and max values) for determining whether the sensor is operating under Normal, NonCritical, Critical, or Fatal conditions. If the CurrentReading is between LowerThresholdCritical and Lower ThresholdFatal, then the CurrentState is Critical.	Will have appropriate value if sensor supports this threshold. If sensor does not support this threshold, this property will not be set.
LowerThresh oldFatal	sint32	The sensor's threshold values specify the ranges (min and max values) for determining whether the sensor is operating under Normal, NonCritical, Critical, or Fatal conditions. If the CurrentReading is below LowerThresholdFatal, then the current state is Fatal.	Will have appropriate value if sensor supports this threshold. If sensor does not support this threshold, this property will not be set.

TABLE: Properties for Oracle_NumericSensor (*Continued*)

Property	Data Type	Description	ILOM Value
Operational Status	uint16[]	<p>The <code>OperationalStatus</code> property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration definitions can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for the enumeration definitions are as follows: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ..., 0x8000..}</p>	Will have appropriate value.
PossibleStates	string	<p><code>PossibleStates</code> enumerates the string outputs of the sensor. For example, a switch sensor can output the states <i>On</i>, or <i>Off</i>. Another implementation of the switch might output the states <i>Open</i> and <i>Close</i>. Another example is a <code>NumericSensor</code> supporting thresholds. This sensor can report the states like Normal, Upper Fatal, Lower Non-Critical, and so on. A <code>NumericSensor</code> that does not publish readings and thresholds, but can store the data internally and still report its states.</p>	Will have appropriate values depending on the type of the sensor.
RateUnits	uint16[]	<p>Specifies if the units returned by this sensor are rate units. All the values returned by this sensor are represented in the units obtained by (<code>BaseUnits</code> * 10 raised to the power of the <code>UnitModifier</code>). This is true unless this property (<code>RateUnits</code>) has a value different from None. For example, if <code>BaseUnits</code> is Volts and the <code>UnitModifier</code> is -6, then the units of the values returned are microvolts. But, if the <code>RateUnits</code> property is set to a value other than "None", then the units are further qualified as rate units. In this example, if <code>RateUnits</code> is set to "Per Second", then the values returned by the Sensor are in microvolts/second. The units apply to all numeric properties of the sensor, unless explicitly overridden by the Units qualifier. Any implementation of <code>CurrentReading</code> should be qualified with either a Counter or a Gauge qualifier, depending on the characteristics of the sensor being modeled.</p>	Will be set to 0.

TABLE: Properties for Oracle_NumericSensor (Continued)

Property	Data Type	Description	ILOM Value
RequestedState	uint16[]	<p>The RequestedState property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for the above definitions are as follows {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, .., 32768..65535}</p> <p>Note - When EnabledState is set to 5 (Not Applicable), then this property has no meaning.</p>	Set to 12 (Not Applicable).
SensorType	uint16[]	<p>Identifies the type of the sensor, for example, voltage sensor or temperature sensor. If the type is set to Other, then the OtherSensorType description can be used to further identify the type, or if the sensor has numeric readings, then the type of the sensor can be implicitly determined by the Units. A description of the different sensor types is as follows:</p> <ul style="list-style-type: none"> • A temperature sensor measures the environmental temperature. • Voltage and current sensors measure electrical voltage and current readings. • A tachometer measures speed/revolutions of a device. For example, a fan device can have an associated tachometer that measures its speed. • A counter is a general purpose sensor that measures some numerical property of a device. • A counter value can be cleared, but it never decreases. • A switch sensor has states like Open or Close, On or Off, or Up or Down. • A Lock has states of Locked or Unlocked. Humidity, smoke detection, and air flow sensors measure the equivalent environmental characteristics. • A presence sensor detects the presence of a PhysicalElement. • A power consumption sensor measures the instantaneous power consumed by a managed element. • A power production sensor measures the instantaneous power produced by a managed element such as a power supply or a voltage regulator. • A pressure sensor is used to report pressure. 	Appropriate value.

TABLE: Properties for Oracle_NumericSensor (*Continued*)

Property	Data Type	Description	ILOM Value
		<p>The following values apply: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, .., 32768..65535}</p> <p>Definitions of these values are: {Unknown, Other, Temperature, Voltage, Current, Tachometer, Counter, Switch, Lock, Humidity, Smoke Detection, Presence, Air Flow, Power Consumption, Power Production, Pressure, DMTF Reserved, Vendor Reserved}</p>	
SupportedThresholds	uint16[]	<p>An array representing the thresholds supported by this sensor.</p> <p>The following values apply: {0, 1, 2, 3, 4, 5}</p> <p>Definitions of these values are: {LowerThresholdNonCritical, UpperThresholdNonCritical, LowerThresholdCritical, UpperThresholdCritical, LowerThresholdFatal, UpperThresholdFatal}</p>	String values of supported thresholds.
UpperThresholdCritical	sint32	The sensor's threshold values specify the ranges (min and max values) for determining whether the sensor is operating under Normal, NonCritical, Critical, or Fatal conditions. If the CurrentReading is above UpperThresholdFatal, then the current state is Fatal.	Will have appropriate value if sensor supports this threshold. If sensor does not support this threshold, then this property will not be set.
UpperThresholdNonCritical	sint32	The sensor's threshold values specify the ranges (min and max values) for determining whether the sensor is operating under Normal, NonCritical, Critical, or Fatal conditions. If the CurrentReading is between LowerThresholdNonCritical and UpperThresholdNonCritical, then the sensor is reporting a normal value. If the CurrentReading is between UpperThresholdNonCritical and UpperThresholdCritical, then the current state is NonCritical.	Will have appropriate value if sensor supports this threshold. If sensor does not support this threshold, then this property will not be set.

Oracle_PhysicalAssetCapabilities

Description:	Provides the capabilities for representing FRU-related information for an associated instance of the <code>CIM_PhysicalElement</code> subclass.
Inheritance:	<code>CIM_PhysicalAssetCapabilities</code>
Properties:	For a description of the supported properties for the <code>Oracle_PhysicalAssetCapabilities</code> class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Physical Asset

TABLE: Properties for Oracle_PhysicalAssetCapabilities

Property	Data Type	Description	ILOM Value
InstanceID	string	<p>The InstanceID property is a mandatory key property.</p> <p>Within the scope of the instantiating Namespace, the InstanceID property uniquely identifies an instance of this class. The value of InstanceID should be constructed using the following preferred algorithm:</p> <p><OrgID> : <LocalID></p> <p>where:</p> <ul style="list-style-type: none"> • <OrgID> and <LocalID> are separated by a colon (:). • <OrgID> must include a copyrighted, trademarked, or otherwise unique name that is owned by the business entity creating or defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority. (This is similar to the <Schema Name>_<Class Name> structure of schema class names.) • <OrgID> must not contain a colon (:). The first colon to appear in InstanceID must appear between <OrgID> and <LocalID>. • <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements. • For DMTF defined instances, the <OrgID> must be set to CIM. <p>If this preferred algorithm is not used, the defining entity must ensure that the resultant InstanceID is not re-used across any instance IDs produced by this or other providers for this instance's Namespace.</p>	Implementation dependent value representing unique ID of PhysicalAssetCapabilities.
FRUInfoSupported	boolean	A boolean that indicates whether the PartNumber, Serial Number, Model, Manufacturer, and SKU properties of PhysicalElement are non-null, non-blank values, and the availability of the complete FRU information.	Will be set to TRUE or FALSE depending on whether the associated instance of CIM_PhysicalElement is considered to be a FRU by the platform.

Oracle_PhysicalComponent

Description:	The PhysicalComponent class represents any low-level or basic component within a package. A component object either can not or does not need to be decomposed into its constituent parts.
Inheritance:	CIM_PhysicalComponent
Properties:	<p>For a description of the supported properties for the Oracle_PhysicalComponent class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Physical Asset

TABLE: Properties for Oracle_PhysicalComponent

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_PhysicalComponent
Tag	string	The Tag property is a mandatory key property. An arbitrary string that uniquely identifies the physical element and serves as the key of the element. The Tag property can contain information such as asset tag or serial number data. The key for the physical element is placed very high in the object hierarchy in order to independently identify the hardware or entity, regardless of physical placement in or on cabinets, adapters, and so on. For example, a hot-swappable or removable component can be taken from its containing (scoping) package and be temporarily unused. The object still continues to exist and can even be inserted into a different scoping container. Therefore, the key for physical element is an arbitrary string and is defined independently of any placement or location-oriented hierarchy.	Set to component NAC name.
-CanBeFRUed	boolean	A boolean that indicates whether this physical element can be a FRU (TRUE) or not (FALSE).	Will be set to TRUE or FALSE depending on whether the component is considered to be a FRU by the platform.
Description	string	The Description property provides a textual description of the object.	Will have appropriate description.

TABLE: Properties for Oracle_PhysicalComponent (Continued)

Property	Data Type	Description	ILOM Value
ElementName	string	<p>User-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Set to component NAC name.
HealthState	uint16	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The possible values are 0 to 30, where 5 means the element is entirely healthy and 30 means the element is completely non-functional. The following continuum is defined:</p> <ul style="list-style-type: none"> • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. • 25 (Critical Failure) - The element is nonfunctional and recovery might not be possible. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. 	Will have the appropriate value depending on whether the component is in error state or not.

TABLE: Properties for Oracle_PhysicalComponent (Continued)

Property	Data Type	Description	ILOM Value
		<ul style="list-style-type: none">• 0 (Unknown) - The implementation cannot report on HealthState at this time. DMTF has reserved the unused portion of the continuum for additional health states in the future. Possible values are: {0, 5, 10, 15, 20, 25, 30, ..} Definitions for these values are: {Unknown, OK, Degraded/Warning, Minor failure, Major failure, Critical failure, Non-recoverable error, DMTF Reserved}	
Manufacturer	string	The name of the organization responsible for producing the PhysicalElement. This organization might be the entity from whom the element is purchased, but this is not necessarily true. The latter information is contained in the Vendor property of CIM_Product.	Will have the appropriate value if the component is considered as a FRU by the platform.
Model	string	The name by which the PhysicalElement is generally known.	Will have the appropriate value if the component is considered a FRU by the platform.

TABLE: Properties for Oracle_PhysicalComponent (Continued)

Property	Data Type	Description	ILOM Value
OperationalStatus	Uint16 []	<p>Indicates the current statuses of the element. Various operational statuses are defined. Many of the enumeration's values are self-explanatory. However, a few are not and are described here in more detail.</p> <ul style="list-style-type: none"> • Stressed - indicates that the element is functioning, but needs attention. Examples of stressed states are overload, overheated, and so on. • Predictive Failure - indicates that an element is functioning nominally but predicting a failure in the near future. • In Service - describes an element being configured, maintained, cleaned, or otherwise administered. • No Contact - indicates that the monitoring system has knowledge of this element, but has never been able to establish communications with it. • Lost Communication - indicates that the ManagedSystemElement is known to exist and has been contacted successfully in the past, but is currently unreachable. • Stopped and Aborted - are similar, although the former implies a clean and orderly stop, while the latter implies an abrupt stop where the state and configuration of the element might need to be updated. • Dormant - indicates that the element is inactive or quiesced. • Supporting Entity in Error - indicates that this element might be OK but that another element, on which it is dependent, is in error. An example is a network service or endpoint that cannot function due to lower-layer networking problems. 	OperationalStatus[0] will have appropriate value depending on whether the component is in error state or not.

TABLE: Properties for Oracle_PhysicalComponent (Continued)

Property	Data Type	Description	ILOM Value
		<ul style="list-style-type: none">Completed - indicates that the element has completed its operation. This value should be combined with either OK, Error, or Degraded so that a client can tell if the complete operation Completed with OK (passed), Completed with Error (failed), or Completed with Degraded (the operation finished, but it did not complete OK or did not report an error).Power Mode - indicates that the element has additional power mode information contained in the PowerManagementService association. <p>OperationalStatus replaces the Status property on ManagedSystemElement to provide a consistent approach to enumerations, to address implementation needs for an array property, and to provide a migration path from today's environment to the future. This change was not made earlier because it required the deprecated qualifier. Due to the widespread use of the existing Status property in management applications, providers or instrumentation should provide both the Status and OperationalStatus properties. Further, the first value of OperationalStatus should contain the primary status for the element. When instrumented, Status (because it is single-valued) should also provide the primary status of the element.</p> <p>Possible values are:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p> <p>Definitions of these values are:</p> <p>{Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non- Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p>	
PartNumber	string	The part number assigned by the organization that is responsible for producing or manufacturing the PhysicalElement.	Will have the appropriate value if the component is considered a FRU by the platform.

TABLE: Properties for Oracle_PhysicalComponent (Continued)

Property	Data Type	Description	ILOM Value
SKU	string	The stock-keeping unit number for this PhysicalElement.	Will have the appropriate value if the component is considered a FRU by the platform.
SerialNumber	string	A manufacturer-allocated number used to identify the PhysicalElement.	Will have the appropriate value if the component is considered a FRU by the platform.
StatusDescriptions	string[]	<p>Strings describing the various OperationalStatus array values. For example, if Stopping is the value assigned to OperationalStatus, then this property can contain an explanation as to why an object is being stopped.</p> <p>Note - Entries in this array are correlated with those at the same array index in OperationalStatus.</p>	StatusDescriptions[0] will have appropriate description on the reason for the value of OperationalStatus[0].

Oracle_PhysicalElementCapabilities

Description:	Oracle_PhysicalElementCapabilities is used to associate an instance of CIM_PhysicalElement to its capabilities, Oracle_PhysicalAssetCapabilities.
Inheritance:	CIM_ElementCapabilities
Properties:	<p>For a description of the supported properties for the Oracle_PhysicalElementCapabilities class, see the following table.</p> <p>Note - For more details about Oracle’s Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Physical Asset

TABLE: Properties for Oracle_PhysicalElementCapabilities

Property	Data Type	Description	ILOM Value
Capabilities	Oracle_PhysicalAssetCapabilities REF	The Capabilities property is a mandatory key property. The Capabilities object associated with the element.	Object path to an instance of Oracle_PhysicalAssetCapabilities.
ManagedElement	CIM_PhysicalElement REF	The ManagedElement property is a mandatory key property. Identifies the managed element.	Object path to an instance of Oracle_PhysicalElement.



Oracle_PhysicalMemory

Description:	The Oracle_PhysicalMemory is used to represent low-level memory devices such as SIMMs, DIMMs, raw memory chips, and so forth.
Inheritance:	CIM_PhysicalMemory
Properties:	For a description of the supported properties for the Oracle_PhysicalMemory class, see the following table. Note - For more details about Oracle’s Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Physical Asset

TABLE: Properties for Oracle_PhysicalMemory

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_PhysicalMemory.
Tag	string	The Tag property is a mandatory key property. The Tag property is an arbitrary string that uniquely identifies the physical element and serves as the key of the element. The Tag property can contain information such as asset tag or serial number data. The key for PhysicalElement is placed very high in the object hierarchy in order to independently identify the hardware or entity, regardless of physical placement in or on cabinets, adapters, and so on. For example, a hot-swappable or removable component can be taken from its containing (scoping) package and be temporarily unused. The object still continues to exist and can even be inserted into a different scoping container. Therefore, the key for PhysicalElement is an arbitrary string and is defined independently of any placement or location-oriented hierarchy.	Set to component NAC name.
CanBeFRUed	boolean	The boolean that indicates whether this PhysicalElement is a FRU (TRUE) or not (FALSE).	Set to TRUE or FALSE depending on whether the component is considered to be a FRU by the platform.
Description	string	Textual description of the object.	Appropriate description.
FormFactor	uint16[]	The implementation form factor for the chip. For example, values such as SIMM (7), TSOP (9) or PGA (10) can be specified. The following values apply: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23} Definitions for these values are: {Unknown, Other, SIP, DIP, ZIP, SOJ, Proprietary, SIMM, DIMM, TSOP, PGA, RIMM, SODIMM, SRIMM, SMD, SSMP, QFP, TQFP, SOIC, LCC, PLCC, BGA, FPBGA, LGA}	Set to value 8 (DIMM).

TABLE: Properties for Oracle_PhysicalMemory (*Continued*)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Will have appropriate value depending on whether the component is in error state or not.
MemoryType	uint16[]	<p>The type of PhysicalMemory. Synchronous DRAM is also known as SDRAM. Cache DRAM is also known as CDRAM. CDRAM is also known as Cache DRAM. SDRAM is also known as Synchronous DRAM. BRAM is also known as Block RAM.</p> <p>The following values apply:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26..32567, 32568..65535}</p> <p>Definitions for these values are:</p> <p>{Unknown, Other, DRAM, Synchronous DRAM, Cache DRAM, EDO, EDRAM, VRAM, SRAM, RAM, ROM, Flash, EEPROM, FEPRAM, EPROM, CDRAM, 3DRAM, SDRAM, SGRAM, RDRAM, DDR, DDR-2, BRAM, FB-DIMM, DDR3, FBD2, DMTF Reserved, Vendor Reserved}</p>	Appropriate value.

TABLE: Properties for Oracle_PhysicalMemory (Continued)

Property	Data Type	Description	ILOM Value
Manufacturer	string	The name of the organization responsible for producing the PhysicalElement. This organization might be the entity from whom the Element is purchased, but this is not necessarily true. The latter information is contained in the Vendor property of CIM_Product.	Will have appropriate value if the processor chip is considered a FRU by the platform.
Model	string	The name by which the PhysicalElement is generally known.	Will have appropriate value if the processor chip is considered a FRU by the platform.
OperationalStatus	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration values can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Possible values for the enumeration values include: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	OperationalStatus[0] will have appropriate value depending on whether the component is in error state or not.
PartNumber	string	Part number assigned by the organization that is responsible for producing or manufacturing the physical element.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SKU	string	The stock-keeping unit number for this physical element.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SerialNumber	string	A manufacturer-allocated number used to identify the physical element.	Will have appropriate value if the processor chip is considered a FRU by the platform.
StatusDescriptions	string[]	Strings describing the various OperationalStatus array values. For example, if Stopping is the value assigned to OperationalStatus, then this property might contain an explanation as to why an object is being stopped. Note that entries in this array are correlated with those at the same array index in OperationalStatus.	StatusDescriptions[0] will have appropriate description on the reason for the value of OperationalStatus[0].

Oracle_PhysicalPackage

Description:	The <code>Oracle_PhysicalPackage</code> class represents physical elements that contain or host other components.
Inheritance:	<code>CIM_PhysicalPackage</code>
Properties:	<p>For a description of the supported properties for the <code>Oracle_PhysicalPackage</code> class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Physical Asset

TABLE: Properties for Oracle_PhysicalPackage

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_PhysicalPackage.
Tag	string	The Tag property is a mandatory key property. The Tag property is an arbitrary string that uniquely identifies the physical element and serves as the key of the element. The Tag property can contain information such as asset tag or serial number data. The key for PhysicalElement is placed very high in the object hierarchy in order to independently identify the hardware or entity, regardless of physical placement in or on cabinets, adapters, and so on. For example, a hot-swappable or removable component can be taken from its containing (scoping) package and be temporarily unused. The object still continues to exist and can even be inserted into a different scoping container. Therefore, the key for PhysicalElement is an arbitrary string and is defined independently of any placement or location-oriented hierarchy.	Set to component NAC name.
CanBeFRUed	boolean	A boolean that indicates whether this PhysicalElement is a FRU (TRUE) or not (FALSE).	Will be set to TRUE or FALSE depending on whether the component is considered to be a FRU by the platform.
Description	string	Textual description of the object.	Appropriate description.
ElementName	string	The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information. Note that the Name property of ManagedSystemElement is also defined as a user-friendly name. But it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.	Set to component NAC name.

TABLE: Properties for Oracle_PhysicalPackage (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply.</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Will have appropriate value depending on whether the component is in error state or not.
Manufacturer	string	The name of the organization responsible for producing the PhysicalElement. This organization might be the entity from whom the element is purchased, but this is not necessarily true. The latter information is contained in the Vendor property of CIM_Product.	Will have appropriate value if the processor chip is considered as a FRU by the platform.
Model	string	The name by which the PhysicalElement is generally known.	Will have appropriate value if the component is considered as a FRU by the platform.

TABLE: Properties for Oracle_PhysicalPackage (*Continued*)

Property	Data Type	Description	ILOM Value
Operational Status	Uint16 []	<p>Indicates the current statuses of the element. Various operational statuses are defined. Many of the enumeration's values are self-explanatory. However, a few are not and are described here in more detail.</p> <ul style="list-style-type: none">• Stressed - indicates that the element is functioning, but needs attention. Examples of stressed states are overload, overheated, and so on.• Predictive Failure - indicates that an element is functioning nominally but predicting a failure in the near future.• In Service - describes an element being configured, maintained, cleaned, or otherwise administered.• No Contact - indicates that the monitoring system has knowledge of this element, but has never been able to establish communications with it.• Lost Communication - indicates that the ManagedSystemElement is known to exist and has been contacted successfully in the past, but is currently unreachable.• Stopped and Aborted - are similar, although the former implies a clean and orderly stop, while the latter implies an abrupt stop where the state and configuration of the element might need to be updated.• Dormant - indicates that the element is inactive or quiesced.• Supporting Entity in Error - indicates that this element might be OK but that another element, on which it is dependent, is in error. An example is a network service or endpoint that cannot function due to lower-layer networking problems.• Completed - indicates that the element has completed its operation. This value should be combined with either OK, Error, or Degraded so that a client can tell if the complete operation Completed with OK (passed), Completed with Error (failed), or Completed with Degraded (the operation finished, but it did not complete OK or did not report an error).	OperationalStatus[0] will have appropriate value depending on whether the component is in error state or not.

TABLE: Properties for Oracle_PhysicalPackage (Continued)

Property	Data Type	Description	ILOM Value
		<ul style="list-style-type: none">• Power Mode - indicates that the element has additional power mode information contained in the <code>PowerManagementService</code> association. <p><code>OperationalStatus</code> replaces the <code>Status</code> property on <code>ManagedSystemElement</code> to provide a consistent approach to enumerations, to address implementation needs for an array property, and to provide a migration path from today's environment to the future. This change was not made earlier because it required the deprecated qualifier. Due to the widespread use of the existing <code>Status</code> property in management applications, providers or instrumentation should provide both the <code>Status</code> and <code>OperationalStatus</code> properties. Further, the first value of <code>OperationalStatus</code> should contain the primary status for the element. When instrumented, <code>Status</code> (because it is single-valued) should also provide the primary status of the element.</p> <p>Possible values are:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p> <p>Definitions of these values are:</p> <p>{Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non- Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p>	

TABLE: Properties for Oracle_PhysicalPackage (Continued)

Property	Data Type	Description	ILOM Value
PackageType	uint16[]	<p>Enumeration defining the type of the PhysicalPackage. Note that this enumeration expands on the list in the Entity MIB (the attribute, entPhysicalClass). The numeric values are consistent with the CIM enumeration numbering guidelines, but are slightly different from the MIB values.</p> <ul style="list-style-type: none">• Unknown - indicates that the package type is not known.• Other - indicates that the package type does not correspond to an existing enumerated value. The value is specified using the OtherPackageType property.• Rack through Port or Connector - these values are defined per the Entity-MIB (where the semantics of rack are equivalent to the MIB's stack value).• The other values (for battery, processor, memory, power source or generator and storage media package) are self-explanatory. <p>A value of the blade server should be used when the PhysicalPackage contains the operational hardware aspects of a ComputerSystem, without the supporting mechanicals such as power and cooling. For example, a blade server (server module) includes processors and memory, and relies on the containing chassis to supply power and cooling. In many respects, a blade can be considered a module or card. However, it is tracked differently by inventory systems and differs in terms of service philosophy. For example, a blade server is intended to be hot-plugged into a hosting enclosure without requiring additional cabling, and does not require a cover to be removed from the enclosure for installation.</p>	Appropriate value.

TABLE: Properties for Oracle_PhysicalPackage (Continued)

Property	Data Type	Description	ILOM Value
		Similarly, a blade expansion module has characteristics of a blade server and a module or card. However, it is distinct from both due to inventory tracking and service philosophy, and because of its hardware dependence on a blade. A blade expansion module (or card) must be attached to a blade before you insert the resultant assembly into an enclosure. The following values apply: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17} Definitions for these values are: {Unknown, Other, Rack, Chassis/Frame, Cross Connect/Backplane, Container/Frame Slot, Power Supply, Fan, Sensor, Module/Card, Port/Connector, Battery, Processor, Memory, Power Source/Generator, Storage Media Package (for example, Disk or Tape Drive), Blade, Blade Expansion}	
PartNumber	string	Part number assigned by the organization that is responsible for producing or manufacturing the PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SKU	string	The SKU property is a manufacturer-allocated number used to identify the PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SerialNumber	string	A manufacturer-allocated number used to identify the PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
StatusDescriptions	string[]	Strings describing the various OperationalStatus array values. For example, if Stopping is the value assigned to OperationalStatus, then this property can contain an explanation as to why an object is being stopped. Note that entries in this array are correlated with those at the same array index in OperationalStatus.	StatusDescriptions[0] will have appropriate description on the reason for the value of OperationalStatus[0].

Oracle_Processor

Description:	Identifies capabilities and management of the processor logical device.
Inheritance:	CIM_Processor
Properties:	For a description of the supported properties for the Oracle_Processor class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	CPU

TABLE: Properties for Oracle_Processor

Property	Data Type	Description	ILOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_Processor.
DeviceID	string	The DeviceID property is a mandatory key property. The Device ID indicates an address or other identifying information used to uniquely name the LogicalDevice.	Will be set to the NAC name of the sensor.
SystemCreationClassName	string	The SystemCreationName property is a mandatory key property. Indicates the CreationClassName of the scoping system.	Set to Oracle_ComputerSystem.
SystemName	string	The SystemName property is a mandatory key property. Indicates the SystemName of the scoping system.	Set to Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem that represents the controllee.

TABLE: Properties for Oracle_Processor (*Continued*)

Property	Data Type	Description	ILOM Value
CPUStatus	uint16[]	<p>Indicates the current status of the processor. For example, the processor might be disabled by the user (value=2), or disabled due to a POST error (value=3). Information in this property can be obtained from SMBIOS, the type 4 structure, and the status attribute.</p> <p>The following values are apply: {0, 1, 2, 3, 4, 7}</p> <p>Definitions of these values are: {Unknown, CPU Enabled, CPU Disabled by User, CPU Disabled By BIOS (POST Error), CPU Is Idle, Other}</p>	Appropriate value.
ElementName	string	<p>The ElementName property is a user-friendly name.</p> <p>This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note that the Name property of ManagedSystemElement is also defined as a user-friendly name. But it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Will be set to the NAC name of the sensor.
EnabledDefault	uint16[]	<p>Enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is Enabled (value=2).</p> <p>The following values apply: {2, 3, 5, 6, 7, 9, ..., 32768..65535}</p> <p>Definitions for these values are: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p>	Set to default value 2 (Enabled).

TABLE: Properties for Oracle_Processor (*Continued*)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, shutting down (value=4) and starting (value=10) are transient states between enabled and disabled. The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown)• 1 (Other)• 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests.• 3 (Disabled) - The element will not execute commands and will drop any new requests.• 4 (Shutting Down) - The element is in the process of going to a disabled state.• 5 (Not Applicable) - The element does not support being enabled or disabled.• 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests.• 7 (Test) - The element is in a test state.• 8 (Deferred) - The element might be completing commands, but will queue any new requests.• 9 (Quiesce) - The element is enabled but in a restricted mode.• 10 (Starting) - The element is in the process of going to an enabled state. New requests are queued.• 11..32767 (DMTF Reserved)• 32768..65535 (Vendor Reserved)	Appropriate value.

TABLE: Properties for Oracle_Processor (Continued)

Property	Data Type	Description	ILOM Value
Family	uint16[]	<p>The Processor family type. For example, values include <i>Pentium(R) processor with MMX(TM) technology</i> (value=14) and <i>68040</i> (value=96).</p> <p>The following values apply:</p> <p>{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 48, 49, 50, 51, 52, 53, 54, 55, 64, 65, 66, 67, 68, 69, 80, 81, 82, 83, 84, 85, 86, 87, 88, 96, 97, 98, 99, 100, 101, 112, 120, 121, 122, 128, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 160, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 200, 201, 202, 203, 204, 210, 211, 212, 213, 230, 250, 251, 254, 255, 260, 261, 280, 281, 300, 301, 302, 320, 350, 500, 65534, 65535}</p>	Appropriate value.

TABLE: Properties for Oracle_Processor (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none">• 0 (Unknown) - The implementation cannot report on HealthState at this time.• 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error.• 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors.• 15 (Minor Failure) - All functionality is available but some might be degraded.• 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working.• 25 (Critical Failure) - The element is non-functional and recovery might not be possible.• 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Appropriate value.

TABLE: Properties for Oracle_Processor (*Continued*)

Property	Data Type	Description	ILOM Value
OperationalStatus	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory. Enumeration definitions can include any of the following:</p> <p>{Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for the enumeration definition are as follows:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, .., 0x8000..}</p>	Appropriate value.
RequestedState	uint16[]	<p>The RequestedState property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following:</p> <p>{Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions are:</p> <p>{0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, .., 32768..65535}</p> <p>Note - When EnabledState is set to 5 (Not Applicable), then this property has no meaning.</p>	Set to 12 (Not Applicable).

Oracle_ProcessorChip

Description:	Identifies the integrated circuit hardware for the processor.
Inheritance:	CIM_Chip
Properties:	<p>For a description of the supported properties for the <code>Oracle_ProcessorChip</code> class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Physical Asset

TABLE: Properties for Oracle_ProcessorChip

Property	Data Type	Description	ILOM Value
CreationClassName	string	<p>The CreationClassName property is a mandatory key property.</p> <p>CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.</p>	Set to Oracle_ProcessorChip.
Tag	string	<p>The Tag property is a mandatory key property.</p> <p>The Tag property is an arbitrary string that uniquely identifies the physical element and serves as the key of the element. The Tag property can contain information such as asset tag or serial number data. The key for PhysicalElement is placed very high in the object hierarchy in order to independently identify the hardware or entity, regardless of physical placement in or on cabinets, adapters, and so on. For example, a hot-swappable or removable component can be taken from its containing (scoping) package and be temporarily unused. The object still continues to exist and can even be inserted into a different scoping container. Therefore, the key for PhysicalElement is an arbitrary string and is defined independently of any placement or location-oriented hierarchy.</p>	Set to component NAC name.
CanBeFRUed	boolean	The boolean indicates whether this PhysicalElement can be a FRU (TRUE) or not (FALSE).	Set to TRUE or FALSE depending on whether the component is considered to be a FRU by the platform.
Description	string	Textual description of the object.	Appropriate description.
ElementName	string	<p>The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note that the Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Set to component NAC name.

TABLE: Properties for Oracle_ProcessorChip (*Continued*)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Will have appropriate value depending on whether the component is in error state or not.
Manufacturer	string	The name of the organization responsible for producing the physical element. This organization might be the entity from whom the element is purchased, but this is not necessarily true. The latter information is contained in the <code>Vendor</code> property of <code>CIM_Product</code> .	Will have appropriate value if the processor chip is considered a FRU by the platform.
Model	string	The name by which the physical element is generally known.	Will have appropriate value if the processor chip is considered a FRU by the platform.

TABLE: Properties for Oracle_ProcessorChip (*Continued*)

Property	Data Type	Description	ILOM Value
Operational Status	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory. Enumeration definitions can include any of the following:</p> <p>{Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ..., 0x8000..}</p>	OperationalStatus[0] will have appropriate value depending on whether the component is in error state or not.
PartNumber	string	Part number assigned by the organization that is responsible for producing or manufacturing the PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SKU	string	The stock-keeping unit number for this PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
SerialNumber	string	A manufacturer-allocated number used to identify this PhysicalElement.	Will have appropriate value if the processor chip is considered a FRU by the platform.
StatusDescriptions	string[]	Strings describing the various OperationalStatus array values. For example, if Stopping is the value assigned to OperationalStatus, then this property can contain an explanation as to why an object is being stopped. Note that entries in this array are correlated with those at the same array index in OperationalStatus.	StatusDescriptions[0] will have appropriate description on the reason for the value of OperationalStatus[0].

Oracle_Realizes

Description:	Oracle_Realizes is the association that defines the mapping between LogicalDevices and the PhysicalElements that implement them.
Inheritance:	CIM_Realizes
Properties:	<p>For a description of the supported properties for the Oracle_Realizes class, see the following table.</p> <p>Note - For more details about Oracle’s Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	<ul style="list-style-type: none">• Physical Asset• CPU• System Memory

TABLE: Properties for Oracle_Realizes

Property	Data Type	Description	ILOM Value
Antecedent	CIM_PhysicalElement REF	The Antecedent property is a mandatory key property. The physical component that implements the device.	Object path to an instance of CIM_PhysicalElement.
Dependent	CIM_LogicalDevice REF	The Dependent property is a mandatory key property. The LogicalDevice.	Object path to an instance of CIM_LogicalDevice.

Oracle_RegisteredProfile

Description:	Provides implementation conformance to a CIM profile.
Inheritance:	CIM_RegisteredProfile
Properties:	<p>For a description of the supported properties for the Oracle_RegisteredProfile class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	Profile Registration

TABLE: Properties for Oracle_RegisteredProfile

Property	Data Type	Description	ILOM Value
InstanceID	string	<p>The InstanceID property is a key mandatory property.</p> <p>Within the scope of the instantiating Namespace, the InstanceID property uniquely identifies an instance of this class. The value of InstanceID should be constructed using the following preferred algorithm:</p> <p><OrgID>:<LocalID></p> <p>where:</p> <ul style="list-style-type: none"> • <OrgID> and <LocalID> are separated by a colon (:). • <OrgID> must include a copyrighted, trademarked, or otherwise unique name that is owned by the business entity creating or defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority. (This is similar to the <Schema Name>_<Class Name> structure of schema class names.) • <OrgID> must not contain a colon (:). The first colon to appear in InstanceID must appear between <OrgID> and <LocalID> . • <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements • For DMTF defined instances, the <OrgID> must be set to CIM. <p>If this preferred algorithm is not used, the defining entity must ensure that the resultant InstanceID is not re-used across any instance IDs produced by this or other providers for this instance's Namespace.</p>	Implementation dependent value representing unique ID.
AdvertiseTypes	uint16[]	<p>Signifies the advertisement for the profile information. It is used by the advertising services of the WBEM infrastructure to determine what should be advertised, using what mechanisms. The property is an array so that the profile might be advertised using several mechanisms.</p> <p>Note - If this property is null/uninitialized, this is equivalent to specifying the value 2 (Not Advertised).</p>	Will have the value 2 (Not Advertised).

TABLE: Properties for Oracle_RegisteredProfile (*Continued*)

Property	Data Type	Description	ILOM Value
RegisteredName	string	The name of this registered profile. Since multiple versions can exist for the same RegisteredName, the combination of RegisteredName, RegisteredOrganization, and RegisteredVersion must uniquely identify the registered profile within the scope of the organization.	Value of supported profile name.
RegisteredOrganization	uint16[]	<p>The organization that defines this profile.</p> <p>The values for this property include: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ..}</p> <p>Definitions for these values are as follows: {Other, DMTF, CompTIA, Consortium for Service Innovation, FAST, GGF, INTAP, itSMF, NAC, Northwest Energy Efficiency Alliance, SNIA, TM Forum, The Open Group, ANSI, IEEE, IETF, INCITS, ISO, W3C, OGF, DMTF Reserved}</p>	Will have the value 2 (DMTF).
RegisteredVersion	string	<p>The version of this profile. The string representing the version must be in the form: M + . + N + . + U where:</p> <ul style="list-style-type: none"> • M - The major version (in numeric form) describing the profile's creation or last modification. • N - The minor version (in numeric form) describing the profile's creation or last modification. • U - The update (for example, errata, patch, and so forth, in numeric form) describing the profile's creation or last modification. 	Will have, for example, 1.0.0 as the value.

Oracle_RecordLog

Description:	Oracle_RecordLog serves as an aggregation point for log entry objects. It is used to represent the IPMI SEL log. Properties of Oracle_RecordLog follow guidelines in IPMI CIM Mapping Guideline.
Inheritance:	CIM_RecordLog
Properties:	For a description of the supported properties for the Oracle_RecordLog class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Record Log

TABLE: Properties for Oracle_RecordLog

Property	Data Type	Description	ILOM Value
InstanceID	string	<p>The InstanceID property is a key manadatory property. Within the scope of the instantiating NameSpace, the InstanceID property uniquely identifies an instance of this class. The value of InstanceID should be constructed using the following preferred algorithm:</p> <p><OrgID>:<LocalID></p> <p>where:</p> <ul style="list-style-type: none">• <OrgID> and <LocalID> are separated by a colon (:).• <OrgID> must include a copyrighted, trademarked or otherwise unique name that is owned by the business entity creating or defining the InstanceID, or is a registered ID that is assigned to the business entity by a recognized global authority. (This is similar to the <Schema Name>_<Class Name> structure of schema class names.)• <OrgID> must not contain a colon (:). The first colon to appear in InstanceID must appear between <OrgID> and <LocalID>.• <LocalID> is chosen by the business entity and should not be re-used to identify different underlying (real-world) elements.• For DMTF defined instances, the <OrgID> must be set to CIM. <p>If this preferred algorithm is not used, the defining entity must ensure that the resultant InstanceID is not re-used across any instance IDs produced by this or other providers for this instance's NameSpace.</p>	Implementation-dependent value representing unique ID.
CurrentNumberOfRecords	UInt64	Current number of records in the log.	Appropriate value.
ElementName	string	<p>The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information.</p> <p>Note - The Name property of ManagedSystemElement is also defined as a user-friendly name. But, it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.</p>	Will hve the value SEL Log.

TABLE: Properties for Oracle_RecordLog (*Continued*)

Property	Data Type	Description	ILOM Value
EnabledDefault	uint16[]	<p>An enumerated value indicating an administrator's default or startup configuration for the EnabledState of an element. By default, the element is Enabled (value=2). The following values apply: {2, 3, 5, 6, 7, 9, ..., 32768..65535}</p> <p>Definitions of these values are: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}</p>	Will be set to default value 2 (Enabled).
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, shutting down (value=4) and starting (value=10) are transient states between enabled and disabled. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) • 1 (Other) • 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests. • 3 (Disabled) - The element will not execute commands and will drop any new requests. • 4 (Shutting Down) - The element is in the process of going to a disabled state. • 5 (Not Applicable) - The element does not support being enabled or disabled. • 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests • 7 (Test) - The element is in a test state. • 8 (Deferred) - The element might be completing commands, but will queue any new requests. • 9 (Quiesce) - The element is enabled but in a restricted mode. • 10 (Starting) - The element is in the process of going to an enabled state. New requests are queued. • 11..32767 (DMTF Reserved) • 32768..65539 (Vendor Reserved) 	Appropriate value.

TABLE: Properties for Oracle_RecordLog (*Continued*)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Appropriate value.
LogState	uint16[]	<p>LogState is an integer enumeration that indicates the current state of a log represented by CIM_Log subclasses. LogState is to be used in conjunction with the EnabledState property to fully describe the current state of the log. The following text briefly summarizes the various log states: Unknown (0) indicates the state of the log is unknown. Normal (2) indicates that the log is or could be executing logging commands, will process any queued log entries, and will queue new logging requests. Erasing (3) indicates that the log is being erased. Not Applicable (4) indicates the log does not support representing a log state.</p> <p>The following values apply: {0, 2, 3, 4, ..., 32768..65535}</p> <p>Definitions for these values are: {Unknown, Normal, Erasing, Not Applicable, DMTF Reserved, Vendor Reserved}</p>	Appropriate value.
MaxNumberOfRecords	UInt64	Maximum number of records that can be captured in the log. If undefined, a value of 0 should be specified.	Appropriate value.

TABLE: Properties for Oracle_RecordLog (*Continued*)

Property	Data Type	Description	ILOM Value
OperationalStatus	uint16[]	<p>The <code>OperationalStatus</code> property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration definitions can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Values for the enumeration definitions include: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ..., 0x8000..}</p>	Appropriate value.
OverwritePolicy	uint16[]	<p>Integer enumeration that indicates whether the log, represented by the <code>CIM_Log</code> subclasses, can overwrite its entries. Unknown (0) indicates that the log's overwrite policy is unknown. Wraps When Full (2) indicates that the log overwrites its entries with new entries when the log has reached its maximum capacity. Never Overwrites (7) indicates that the log never overwrites its entries by the new entries.</p> <p>The following values apply: {0, 2, 7, ..., 32768..65535}</p> <p>Definitions for these values are: {Unknown, Wraps When Full, Never Overwrites, DMTF Reserved, Vendor Reserved}</p>	Will have value 2 (Wraps When Full).
RequestedState	uint16[]	<p>The <code>RequestedState</code> property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by <code>EnabledState</code>. This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions are as follows: {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ..., 32768..65535}</p> <p>Note - When <code>EnabledState</code> is set to 5 (Not Applicable), then this property has no meaning.</p>	Will bet set to 12 (Not Applicable).

Oracle_ReferencedProfile

Description:	Oracle_ReferencedProfile is used to associate an instance of Oracle_RegisteredProfile to the instance of Oracle_RegisteredProfile representing the Base Server profile. ILOM uses Scoping Class advertisement methodology. See the Profile Registration profile in “Supported DMTF SMASH Profiles and CIM Classes” on page 148 for details.
Inheritance:	CIM_ReferencedProfile
Properties:	For a description of the supported properties for the Oracle_ReferencedProfile class, see the following table. Note - For more details about Oracle’s Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Profile Registration

TABLE: Properties for Oracle_ReferencedProfile

Property	Data Type	Description	ILOM Value
Antecedent	Oracle_RegisteredProfile REF	The Antecedent property is a mandatory key property. Instance of Oracle_RegisteredProfile.	Object path to an instance of Oracle_RegisteredProfile .
Dependent	Oracle_RegisteredProfile REF	The Dependent property is a mandatory key property. Indicates the Oracle_RegisteredProfile.	Object path to the instance of Oracle_RegisteredProfile representing the Base Server profile.

Oracle_Sensor

Description:	Represents a hardware component capable of measuring the characteristics of a physical property (for example, the temperature or voltage characteristics of a computer system).
Inheritance:	CIM_Sensor
Properties:	For a description of the supported properties for the Oracle_Sensor class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Sensor

TABLE: Properties for Oracle_Sensor

Property	Data Type	Description	LOM Value
CreationClassName	string	The CreationClassName property is a mandatory key property. CreationClassName indicates the name of the class or the subclass used in the creation of an instance. When used with the other key properties of this class, this property allows all instances of this class and its subclasses to be uniquely identified.	Set to Oracle_Sensor.
DeviceID	string	The DeviceID property is a mandatory key property. The DeviceID property indicates an address or other identifying information used to uniquely name the LogicalDevice.	Set to the NAC name of the sensor.
SystemCreationClassName	string	The SystemCreationClassName property is a mandatory key property. Indicates the SystemCreationClassName for the scoping system.	Set to Oracle_ComputerSystem.

TABLE: Properties for Oracle_Sensor (Continued)

Property	Data Type	Description	ILOM Value
SystemName	string	The SystemName property is a mandatory key property. Indicates the SystemName of the scoping system.	Set to Oracle_ComputerSystem.Name of the instance of Oracle_ComputerSystem that represents the controllee.
CurrentState	string	The current state indicated by the sensor. This is always one of the PossibleStates.	Value representing current state of the sensor.
ElementName	string	The ElementName property is a user-friendly name. This property allows each instance to define a user-friendly name in addition to its key properties, identity data, and description information. Note that the Name property of ManagedSystemElement is also defined as a user-friendly name. But it is often subclassed to be a key. It is not reasonable that the same property can convey both identity and a user-friendly name, without inconsistencies. Where Name exists and is not a key (such as for instances of LogicalDevice), the same information can be present in both the Name and ElementName properties.	Will be set to the NAC name of the sensor.
EnabledDefault	uint16[]	An enumerated value indicating an administrator's default or startup configuration for the enabled state of an element. By default, the element is Enabled (value=2). The following values apply: {2, 3, 5, 6, 7, 9, ..., 32768..65535} Definitions for these values are: {Enabled, Disabled, Not Applicable, Enabled but Offline, No Default, Quiesce, DMTF Reserved, Vendor Reserved}.	Set to default value 2 (Enabled).

TABLE: Properties for Oracle_Sensor (Continued)

Property	Data Type	Description	ILOM Value
EnabledState	uint16[]	<p>Integer enumeration that indicates the enabled and disabled states of an element. It can also indicate the transitions between these requested states. For example, shutting down (value=4) and starting (value=10) are transient states between enabled and disabled. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) • 1 (Other) • 2 (Enabled) - The element is or could be executing commands, will process any queued commands, and queues new requests. • 3 (Disabled) - The element will not execute commands and will drop any new requests. • 4 (Shutting Down) - The element is in the process of going to a disabled state. • 5 (Not Applicable) - The element does not support being enabled or disabled. • 6 (Enabled but Offline) - The element might be completing commands, and will drop any new requests. • 7 (Test) - The element is in a test state. • 8 (Deferred) - The element might be completing commands, but will queue any new requests. • 9 (Quiesce) - The element is enabled but in a restricted mode. • 10 (Starting) - The element is in the process of going to an enabled state. New requests are queued. • 11..32767 (DMTF Reserved) • 32768..65535 (Vendor Reserved) 	Will have appropriate value depending on whether the sensor is enabled, disabled, or unknown.

TABLE: Properties for Oracle_Sensor (Continued)

Property	Data Type	Description	ILOM Value
HealthState	uint16[]	<p>Indicates the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents. The following values apply:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The implementation cannot report on HealthState at this time. • 5 (OK) - The element is fully functional and is operating within normal operational parameters and without error. • 10 (Degraded/Warning) - The element is in working order and all functionality is provided. However, the element is not working to the best of its abilities. For example, the element might not be operating at optimal performance or it might be reporting recoverable errors. • 15 (Minor Failure) - All functionality is available but some might be degraded. • 20 (Major Failure) - The element is failing. It is possible that some or all of the functionality of this component is degraded or not working. • 25 (Critical Failure) - The element is non-functional and recovery might not be possible. • 30 (Non-Recoverable Error) - The element has completely failed, and recovery is not possible. All functionality provided by this element has been lost. <p>DMTF has reserved the unused portion of the continuum for additional health states in the future.</p>	Appropriate value.
OperationalStatus	uint16[]	<p>The OperationalStatus property indicates the current statuses of the element.</p> <p>Various operational statuses are defined. Many of the enumeration's values are self-explanatory.</p> <p>Enumeration values can include any of the following: {Unknown, Other, OK, Degraded, Stressed, Predictive Failure, Error, Non-Recoverable Error, Starting, Stopping, Stopped, In Service, No Contact, Lost Communication, Aborted, Dormant, Supporting Entity in Error, Completed, Power Mode, DMTF Reserved, Vendor Reserved}</p> <p>Possible values for the enumeration values include: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ..., 0x8000..}</p>	Appropriate value.

TABLE: Properties for Oracle_Sensor (Continued)

Property	Data Type	Description	ILOM Value
PossibleStates	string	Enumerates the string outputs of the sensor. For example, a switch sensor can output the states On or Off. Another implementation of the switch may output the states Open, and Close. Another example is a NumericSensor supporting thresholds. This sensor can report the states like Normal, Upper Fatal, Lower Non-Critical, and so forth. A NumericSensor that does not publish readings and thresholds, but can store this data internally and still report its states.	Appropriate values depending on the type of the sensor.
RequestedState	uint16[]	<p>The RequestedState property is an integer enumeration that indicates the last requested or desired state for the element, irrespective of the mechanism through which it was requested. The actual state of the element is represented by EnabledState. This property is provided to compare the last requested and current enabled or disabled states.</p> <p>Element definitions include any of the following: {Unknown, Enabled, Disabled, Shut Down, No Change, Offline, Test, Deferred, Quiesce, Reboot, Reset, Not Applicable, DMTF Reserved, Vendor Reserved}</p> <p>Values for these definitions include: {0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ..., 32768..65535}</p> <p>Note - When EnabledState is set to 5 (Not Applicable), then this property has no meaning. Refer to the DMTF CIM EnabledState property description for explanations of the values in the RequestedState enumeration.</p>	Set to 12 (Not Applicable).

TABLE: Properties for Oracle_Sensor (Continued)

Property	Data Type	Description	ILOM Value
SensorType	uint16[]	<p>The type of the sensor, for example, voltage or temperature sensor. If the type is set to Other, then the OtherSensorType description can be used to further identify the type, or if the sensor has numeric readings, then the type of the sensor can be implicitly determined by the units. A description of the different sensor types is as follows:</p> <ul style="list-style-type: none">• A temperature sensor measures the environmental temperature.• Voltage and current sensors measure electrical voltage and current readings.• A tachometer measures speed/revolutions of a device. For example, a fan device can have an associated tachometer which measures its speed.• A counter is a general purpose sensor that measures some numerical property of a device. A counter value can be cleared, but it never decreases.• A switch sensor has states like Open or Close, On or Off, or, Up or Down.• A lock has states of Locked or Unlocked.• Humidity, smoke detection, and air flow sensors measure the equivalent environmental characteristics.• A presence sensor detects the presence of a PhysicalElement.• A power consumption sensor measures the instantaneous power consumed by a managed element.• A power production sensor measures the instantaneous power produced by a managed element such as a power supply or a voltage regulator.• A pressure sensor is used to report pressure. <p>The following values apply: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, .., 32768..65535}</p> <p>Definitions of these values are: {Unknown, Other, Temperature, Voltage, Current, Tachometer, Counter, Switch, Lock, Humidity, Smoke Detection, Presence, Air Flow, Power Consumption, Power Production, Pressure, DMTF Reserved, Vendor Reserved}</p>	Will have appropriate value.

Oracle_SpSystemComponent

Description:	Oracle_SpSystemComponent is used to associate the instance of Oracle_ComputerSystem representing the controllee and the instance of Oracle_ComputerSystem representing the controller.
Inheritance:	CIM_SystemComponent
Properties:	For a description of the supported properties for the Oracle_SpSystemComponent class, see the following table. Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Service Processor

TABLE: Properties for Oracle_SpSystemComponent

Property	Data Type	Description	ILOM Value
GroupComponent	CIM_ComputerSystem REF	The GroupComponent property is a key mandatory property. Indicates the parent system in the association.	Object path to the instance of Oracle_ComputerSystem representing the controllee.
PartComponent	CIM_ComputerSystem REF	The PartComponent property is a key mandatory property. Indicates the child element of a system component.	Object path to the instance of Oracle_ComputerSystem representing the controller.

Oracle_SystemDevice

Description:	Association that represents an explicit relationship in which logical devices are aggregated by a ComputerSystem.
Inheritance:	CIM_SystemDevice
Properties:	<p>For a description of the supported properties for the Oracle_SystemDevice class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	<ul style="list-style-type: none">• Base Server• System Memory• Sensors• CPU• Indicator LED

TABLE: Properties for Oracle_SystemDevices

Property	Data Type	Description	ILOM Value
GroupComponent	Oracle_ComputerSystem REF	The GroupComponent property is a key mandatory property. Indicates the Oracle_ComputerSystem.	Object path to an instance of Oracle_ComputerSystem representing the controllee.
PartComponent	CIM_LogicalDevice REF	The PartComponent property is a key mandatory property. The PartComponent is the LogicalDevice that is a component of a system.	Object path to an instance of CIM_LogicalDevice.

Oracle_ThresholdIndication

Description:	<p>When the client creates an indication subscription in which the filter indicates one of the following:</p> <ul style="list-style-type: none">• CIM_AlertIndication and CIM_AlertIndication.ProbableCause is 52 (Threshold Crossed)• CIM_ThresholdIndication <p>The ILOM CIM sub-system will generate an instance of the Oracle_ThresholdIndication class when it notices a sensor crossing a threshold.</p>
Inheritance:	CIM_ThresholdIndication
Properties:	<p>For a description of the supported properties for the Oracle_ThresholdIndication class, see the following table.</p> <p>Note - For more details about Oracle's Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:</p>
Profile:	None

TABLE: Properties for Oracle_ThresholdIndication

Property	Data Type	Description	ILOM Value
AlertingElementFormat	uint16[]	<p>The format of the AlertingManagedElement property is interpretable based upon the value of this property. Values are defined as:</p> <ul style="list-style-type: none"> • 0 (Unknown) - The format is unknown or not meaningfully interpretable by a CIM client application. • 1 (Other) - The format is defined by the value of the OtherAlertingElementFormat property. • 2 (CIMObjectPath) - The format is a CIMObjectPath, with format <NameSpacePath>:<ClassName>.<Prop1>=<Value1>" , <Prop2>=<Value2>", and so forth, specifying an instance in the CIM schema. <p>The following values apply: {0, 1, 2}</p> <p>Definitions for these values are: {Unknown, Other, CIMObjectPath}</p>	Will have the value 2 (CIMObjectPath).
AlertingManagedElement	string	<p>The identifying information of the entity (that is, the instance) for which this indication is generated. The property contains the path of an instance, encoded as a string parameter, if the instance is modeled in the CIM schema. If not a CIM instance, the property contains some identifying string that names the entity for which the alert is generated. The path or identifying string is formatted per the AlertingElementFormat property.</p>	Will have the string representation of the object path of the sensor that crosses the threshold.

TABLE: Properties for Oracle_ThresholdIndication (*Continued*)

Property	Data Type	Description	ILOM Value
AlertType	uint16[]	<p>Primary classification of the indication. The following values are defined:</p> <ul style="list-style-type: none"> • 1 (Other) - Current indication does not fit into the categories described by this enumeration. • 2 (Communications Alert) - Associated with the procedures and/or processes required to convey information from one point to another. • 3 (Quality of Service Alert) - A degradation or errors in the performance or function of an entity have occurred. • 4 (Processing Error) - A software or processing fault has occurred. • 5 (Device Alert) - An equipment or hardware fault has occurred. • 6 (Environmental Alert) - Refers to an enclosure in which the hardware resides, or other environmental considerations. • 7 (Model Change) - Addresses changes in the information model. For example, it might embed a lifecycle indication to convey the specific model change being alerted. • 8 (Security Alert) - Security violations, detection of viruses, or similar issues have occurred. 	Will have the value 6 (Environmental Alert).
Description	string	Short description for the instance.	Appropriate value describing why the indication is generated.
HardwareComponentObjectPath (Sun-specific)	string	Object path of the associated hardware component.	The object path of an instance of <code>CIM_PhysicalElement</code> .
ObservedValue	string	A string holding the current reading value that exceeds the threshold. This is modeled as a string for universal mapping, similar to the <code>CIM_Sensor</code> properties in the device model.	Appropriate value.

TABLE: Properties for Oracle_ThresholdIndication (*Continued*)

Property	Data Type	Description	LOM Value
ProbableCause	uint16[]	<p>Enumerated value that describes the probable cause of the situation that resulted in the AlertIndication.</p> <p>The following values apply:</p> <p>{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130}</p> <p>Definitions for these values are:</p> <p>{Unknown, Other, Adapter/Card Error, Application Subsystem Failure, Bandwidth Reduced, Connection Establishment Error, Communications Protocol Error, Communications Subsystem Failure, Configuration/Customization Error, Congestion, Corrupt Data, CPU Cycles Limit Exceeded, Dataset/Modem Error, Degraded Signal, DTE-DCE Interface Error, Enclosure Door Open, Equipment Malfunction, Excessive Vibration, File Format Error, Fire Detected, Flood Detected, Framing Error, HVAC Problem, Humidity Unacceptable, I/O Device Error, Input Device Error, LAN Error, Non-Toxic Leak Detected, Local Node Transmission Error, Loss of Frame, Loss of Signal, Material Supply Exhausted, Multiplexer Problem, Out of Memory, Output Device Error, Performance Degraded, Power Problem, Pressure Unacceptable, Processor Problem (Internal Machine Error), Pump Failure, Queue SizeExceeded, Receive Failure, Receiver Failure, Remote NodeTransmission Error, Resource at or Nearing Capacity, ResponseTime Excessive, RetransmissionRate Excessive, Software Error, Software Program AbnormallyTerminated, Software Program Error (Incorrect Results), Storage Capacity Problem, Temperature Unacceptable, Threshold Crossed, Timing Problem, Toxic Leak Detected, Transmit Failure, Transmitter Failure, Underlying Resource Unavailable, Version MisMatch, Previous AlertCleared, Login Attempts Failed, Software Virus Detected, Hardware Security Breached, Denial of Service Detected, Security Credential MisMatch,</p>	Set to 52 (Threshold Crossed).

TABLE: Properties for Oracle_ThresholdIndication (*Continued*)

Property	Data Type	Description	ILOM Value
		Unauthorized Access, Alarm Received, Loss of Pointer, Payload Mismatch, Transmission Error, Excessive Error Rate, Trace Problem, Element Unavailable, Element Missing, Loss of MultiFrame, Broadcast Channel Failure, Invalid Message Received, Routing Failure, Backplane Failure, Identifier Duplication, Protection Path Failure, Sync Lossor Mismatch, Terminal Problem, Real Time Clock Failure, Antenna Failure, Battery Charging Failure, Disk Failure, Frequency Hopping Failure, Loss of Redundancy, Power Supply Failure, Signal Quality Problem, Battery Discharging, Battery Failure, Commercial Power Problem, Fan Failure, Engine Failure, Sensor Failure, Fuse Failure, Generator Failure, Low Battery, Low Fuel, Low Water, Explosive Gas, High Winds, Ice Buildup, Smoke, Memory Mismatch, Out of CPU Cycles, Software Environment Problem, Software Download Failure, Element Reinitialized, Timeout, Logging Problems, Leak Detected, Protection Mechanism Failure, Protecting Resource Failure, Database Inconsistency, Authentication Failure, Breach of Confidentiality, Cable Tamper, Delayed Information, Duplicate Information, Information Missing, Information Modification, Information Out of Sequence, Key Expired, Non-Repudiation Failure, Out of Hours Activity, Out of Service, Procedural Error, Unexpected Information}	
ProviderName	string	The name of the provider generating this indication.	Appropriate value.
SystemCreationClassName	string	The <code>SystemCreationClassName</code> of the scoping system (provider generating this indication).	Will have the value <code>Oracle_ComputerSystem</code> .
SystemName	string	Indicates the <code>SystemName</code> for the scoping system (name for the provider generating this indication).	Will have the value <code>Oracle_ComputerSystem.Name</code> of the instance of <code>Oracle_ComputerSystem</code> representing the controllee.
ThresholdIdentifier	string	Describes the threshold or names the property that represents the threshold, if modeled in the CIM hierarchy. In the latter case, the value should be written as: <code><schema name>_ <class name>.<property name></code> .	Appropriate value.
ThresholdValue	string	Current value of the threshold. This is modeled as a string for universal mapping, similar to the <code>CIM_Sensor</code> properties in the device model.	Appropriate value.

Oracle_UseOfLog

Description:	The Oracle_UseOfLog is used to associate an instance of a Oracle_RecordLog to an instance of the Oracle_ComputerSystem, which represents the controllee.
Inheritance:	CIM_UseOfLog
Properties:	For a description of the supported properties for the Oracle_UseOfLog class, see the following table. Note - For more details about Oracle’s Sun-supported properties (described in the following table), see the DMTF CIM schema, version 2.18.1, at:
Profile:	Record Log Base Server

TABLE: Properties for Oracle_UseOfLog

Property	Data Type	Description	ILOM Value
Antecedent	Oracle_RecordLog REF	The Antecedent property is a mandatory key property. Instance of Oracle_RecordLog	Object path to the instance of Oracle_RecordLog representing the IPMI SEL log.
Dependent	Oracle_ComputerSy stem REF	The Dependent property is a mandatory key property. The Oracle_ComputerSystem.	Object path to the instance of Oracle_ComputerSystem representing the controllee.

SNMP Command Examples

Description	Links
Example SNMP Commands	<ul style="list-style-type: none">• “snmpget Command” on page 263• “snmpwalk Command” on page 264• “snmpbulkwalk Command” on page 265• “snmptable Command” on page 266• “snmpset Command” on page 269• “snmptrapd Command” on page 270

Related Information

- [“SNMP Overview” on page 1](#)

snmpget Command

snmpget -mALL -v1 -cpublic snmp_agent_ip_address sysName.0

As stated in the description of the `sysName.0` MIB object in the SNMPv2-MIB, this command returns an administratively assigned name for this managed node. By convention, this is the node’s fully qualified domain name. If the name is unknown, the value returned is the zero-length string.

For example:

```
% snmpget -v2c -cprivate -mALL snmp_agent_ip_address sysName.0 sysObjectID.0
ilomCtrlDateAndTime.0
SNMPv2-MIB::sysName.0 = STRING: SUNSPHOSTNAME
SNMPv2-MIB::sysObjectID.0 = OID: SUN-ILOM-SMI-MIB::sunILOMSystems
SUN-ILOM-CONTROL-MIB::ilomCtrlDateAndTime.0 = STRING: 2007-12-10,20:33:32.0
```

In addition to the `sysName.0` object, this command displays the content of the `sysObjectID.0` and the `ilomCtrlDateAndTime.0` MIB objects. Notice that the MIB file name is given for each MIB object as part of the reply.

The following descriptions of the MIB objects are taken from the MIB files.

- `sysName` – An administratively assigned name for this managed node. By convention, this is the node’s fully-qualified domain name. If the name is unknown, the value is the zero-length string.
- `sysObjectID` – The vendor’s authoritative identification of the network management subsystem contained in the entity. This value is allocated within the SMI enterprises sub-tree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining ‘what kind of box’ is being managed.
- `ilomCtrlDataAndTime` – The date and time of the device.

snmpwalk Command

The `snmpwalk` command performs a sequence of chained `GETNEXT` requests automatically. It is a work saving command. Rather than having to issue a series of `snmpgetnext` requests, one for each object ID, or node, in a sub-tree, you can simply issue one `snmpwalk` request on the root node of the sub-tree and the command gets the value of every node in the sub-tree.

For example:

```
% snmpwalk -mALL -v1 -cpublic snmp_agent Ip_address system
SNMPv2-MIB::sysDescr.0 = STRING: ILOM machine custom description
SNMPv2-MIB::sysObjectID.0 = OID: SUN-ILOM-SMI-MIB::sunILOMSystems
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (16439826) 1 day, 21:39:58.26
SNMPv2-MIB::sysContact.0 = STRING: set via snmp test
SNMPv2-MIB::sysName.0 = STRING: SUNSPHOSTNAME
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 72
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (14) 0:00:00.14
SNMPv2-MIB::sysORID.1 = OID: IF-MIB::ifMIB
SNMPv2-MIB::sysORID.2 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.3 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.4 = OID: RFC1213-MIB::ip
SNMPv2-MIB::sysORID.5 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.6 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORID.7 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.8 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.9 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
```

```

SNMPv2-MIB::sysORDescr.1 = STRING: The MIB module to describe generic objects
for network interface sub-layers
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB module for SNMPv2 entities
SNMPv2-MIB::sysORDescr.3 = STRING: The MIB module for managing TCP
implementations
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for managing IP and ICMP
implementations
SNMPv2-MIB::sysORDescr.5 = STRING: The MIB module for managing UDP
implementations
SNMPv2-MIB::sysORDescr.6 = STRING: View-based Access Control Model for SNMP.
SNMPv2-MIB::sysORDescr.7 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.8 = STRING: The MIB for Message Processing and
Dispatching.
SNMPv2-MIB::sysORDescr.9 = STRING: The management information definitions for
the SNMP User-based Security Model.
SNMPv2-MIB::sysORUpTime.1 = Timeticks: (1) 0:00:00.01
SNMPv2-MIB::sysORUpTime.2 = Timeticks: (2) 0:00:00.02
SNMPv2-MIB::sysORUpTime.3 = Timeticks: (2) 0:00:00.02
SNMPv2-MIB::sysORUpTime.4 = Timeticks: (2) 0:00:00.02
SNMPv2-MIB::sysORUpTime.5 = Timeticks: (2) 0:00:00.02
SNMPv2-MIB::sysORUpTime.6 = Timeticks: (2) 0:00:00.02
SNMPv2-MIB::sysORUpTime.7 = Timeticks: (14) 0:00:00.14
SNMPv2-MIB::sysORUpTime.8 = Timeticks: (14) 0:00:00.14
SNMPv2-MIB::sysORUpTime.9 = Timeticks: (14) 0:00:00.14

```

snmpbulkwalk Command

The `snmpbulkwalk` command uses the GETBULK SNMP protocol feature to query for an entire tree of information about a network entity. This command can pack more objects into the packets by specifying “repeaters.” As a result, the `snmpbulkwalk` command is faster than the `snmpwalk` command.

Here is example of an `snmpwalk` command with approximate start and end time stamps.

```

% date
Fri Dec 14 12:21:44 EST 2007
% snmpwalk -mALL -v2c -cprivate snmp_agent_IP_address entPhysicalTable>time3
% date
Fri Dec 14 12:21:53 EST 2007

```

Here is example of an `snmpbulkwalk` command performing the same operation. Notice that the `snmpbulkwalk` command is faster than the `snmpwalk` command.

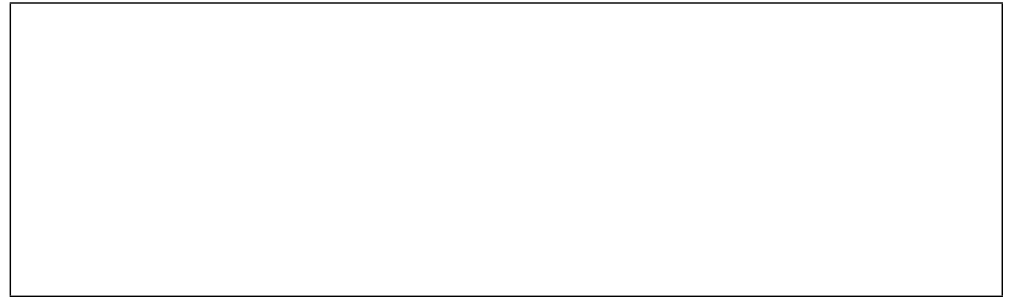
```
% date
Fri Dec 14 12:40:57 EST 2007
% snmpbulkwalk -mALL -v2c -cprivate snmp_agent_ip_address entPhysicalTable>time7
% date
Fri Dec 14 12:41:03 EST 2007
```

snmptable Command

The `snmptable` command retrieves the contents of an SNMP table and displays the contents in a tabular format, that is, one table row at a time, such that the resulting output resembles the table being retrieved. This is contrasted with the `snmpwalk` command, which displays the contents of the table one column at a time.

Here is an example of the `snmptable` command:

```
% snmptable -mALL -v2c -cprivate snmp_agent_ip_address sysORTable
SNMP table: SNMPv2-MIB::sysORTable
sysORID                      sysORDescr                      sysORUpTime
IF-MIB::ifMIB                The MIB module to              0:0:00:00.01
describe generic objects
SNMPv2-MIB::snmpMIB          The MIB module for SNMPv2      0:0:00:00.02
for network interface
TCP-MIB::tcpMIB              The MIB module for            0:0:00:00.02
managing TCP
implementations.
entities.
sub-layers.
RFC1213-MIB::ip              The MIB module for managing    0:0:00:00.02
IP and ICMP implementations.
UDP-MIB::udpMIB              The MIB module for managing    0:0:00:00.02
UDP implementations.
SNMP-VIEW-BASED-ACM-         View-based Access Control      0:0:00:00.02
SNMP-FRAMEWORK-MIB::         The SNMP Management           0:0:00:00.14
MIB::vacmBasicGroup          Model for SNMP.
snmpFrameworkMIB            Architecture MIB.
SNMP-MPD-MIB::snmp           The MIB for Message           0:0:00:00.14
MPDCompliance                Processing and Dispatching.
Compliance
SNMP-USER-BASED-SM-          The management information     0:0:00:00.14
MIB::usnMIBCompliance        definitions for the SNMP
User-based Security Model.
```

Note – While the `snmpget`, `snmpgetnext`, and `snmpwalk` command can be used on any type of MIB object, the `snmptable` command can be used only on MIB table objects. If this command is given any other type of object ID, it will be rejected. This restriction applies to a table entry object, a table column object, and any object that represents information within a table. Only a MIB table object ID can be used with the `snmptable` command.

In the examples of the `snmptable` command, the `-Ci` and `-Cb` options are used. For example, here is an `snmptable` command with the `-Ci` option:

```
% snmptable -Ci -mALL -v2c -cprivate snmp_agent_IP_address
sunPlatFanTable
SNMP table: SUN-PLATFORM-MIB::sunPlatFanTable
index sunPlatFanClass
10                fan
11                fan
17                fan
23                fan
29                fan
30                fan
36                fan
42                fan
```

Here is an example of an `snmptable` command without the `-Ci` option. Notice that the index column is not displayed:

```
% snmptable -mALL -v2c -cprivate snmp_agent_IP_address sunPlatFanTable
SNMP table: SUN-PLATFORM-MIB::sunPlatFanTable
sunPlatFanClass
fan
fan
fan
fan
fan
```

Here is an example of an `snmptable` command with the `-Ci` and `-Cb` options. The output is abbreviated.

```
% snmptable -Ci -Cb -mALL -v2c
-cprivate snmp_agent_IP_address entPhysicalTable
index          Descr          VendorType    ContainedIn
SNMP table: ENTITY      ?SNMPv2-      0            chassis
-MIB::entPhysical      SMI:zeroDotZero
1
Table
```

Here is an example of the same `snmptable` command with the `-Ci` option but without the `-Cb` option. Again the output is abbreviated. Notice that the name of the MIB object is repeated on each heading.

```
% snmptable -Ci -mALL -v2c -cprivate
index          entPhysicalDescr    entPhysical    entPhysical
VendorType      ContainedIn
SNMP table: ENTITY      ?SNMPv2-      0            chassis
1
-MIB::entPhysical      SMI:zeroDotZero
```

Here is another example of an `snmptable` command with both the `-Ci` and `-Cb` options. Notice that the MIB object is not repeated on each heading.

```
% snmptable -Cb -Ci -mALL -v2c -cprivate snmp_agent_IP_address ilomCtrlAlertsTable
SNMP table: SUN-ILOM-CONTROL-MIB::ilomCtrlAlertsTable
in-   Sever-   Type   Destin-   Destin-   SNMPVer-   SNMP-Comm-   Email   Email
dex   ity       email  ation-    ation-    sion       unityOr-    Event   Event
1     criti-    email  ?         0.0.0.0   v1         public     none    none
IP      Email
2-15   dis-     ipmi-   0.0.0.0   ?         v1         public     ?       ?
    able    pet
cal
Filter  Filter
```

Thus, when you used the `-Cb` option with the `snmptable` command, the table output is easier to read.

Here is an example of an `snmptable` command using version 3 of the SNMP protocol:

```
% snmptable -Cb -Ci -mALL -v3 -aMD5 -utestuser -Apassword -lauthNoPriv  
snmp_agent_ip_address sunPlatPowerSupplyTable  
SNMP table: SUN-PLATFORM-MIB::sunPlatPowerSupplyTable  
index sunPlatPowerSupplyClass  
90          powerSupply  
92          powerSupply  
96          powerSupply
```

The following `snmptable` command returns an empty table.

```
% snmptable -Cb -Ci -mALL -v2c -cprivate snmp_agent_ip_address sunPlatBatteryTable  
SUN-PLATFORM-MIB::sunPlatBatteryTable: No entries
```

snmpset Command

While the syntax of the `snmpset` command is similar to that of the `snmpget` command, the commands are quite different. The `snmpget` command merely reads the value of the specified object ID, while the `snmpset` command writes the value specified to the object ID. Further, along with the value to be written to the object ID, you must also specify the data type of the object ID in the `snmpset` command because SNMP objects support more than one data type.

The following example shows how use of the `snmpget` and `snmpset` commands together. The sequence of steps is as follows:

1. Use the `snmpget` command to check to current value of the MIB object.
2. Use the `snmpset` command to change the value of the MIB object.
3. Use the `snmpget` command to verify that the MIB object was in fact changed to the requested value.

```
% snmpget -mALL -v2c -cprivate snmp_agent_ip_address ilomCtrlHttpEnabled.0  
SUN-ILOM-CONTROL-MIB::ilomCtrlHttpEnabled.0 = INTEGER: false(2)  
% snmpset -mALL -v2c -cprivate snmp_agent_ip_address ilomCtrlHttpEnabled.0 i 1  
SUN-ILOM-CONTROL-MIB::ilomCtrlHttpEnabled.0 = INTEGER: true(1)  
% snmpget -mALL -v2c -cprivate snmp_agent_ip_address ilomCtrlHttpEnabled.0  
SUN-ILOM-CONTROL-MIB::ilomCtrlHttpEnabled.0 = INTEGER: true(1)
```

Note that if you try to execute this `snmpset` command using a public community, instead of private, it will not work. This is because the private community has write permission, but the public community does not. The reason code returned by the command does not make this clear because it simply states that the object is not writable.

Here is an example:

```
% snmpset -mALL -v2c -cpublic snmp_agent_ip_address ilomCtrlHttpEnabled.0 i 1
Error in packet.
Reason: notWritable (That object does not support modification)
```

snmptrapd Command

`snmptrapd` is an SNMP application that receives and logs SNMP trap and inform messages. Before your system can receive such messages, you must configure the trap daemon to listen for these messages.

To configure a trap daemon, perform these actions:

1. Configure an SNMP trap destination.

The following example shows how to use the `snmpset` command to configure an `snmptrapd` daemon:

```
% snmpset -mALL -v2c -cprivate snmp_agent_ip_address ilomCtrlAlertSeverity.1 i 2
ilomCtrlAlertType.1 i 2 ilomCtrlAlertDestinationIP.1 a dest_ip_address
SUN-ILOM-CONTROL-MIB::ilomCtrlAlertSeverity.1 = INTEGER: critical(2)
SUN-ILOM-CONTROL-MIB::ilomCtrlAlertType.1 = INTEGER: snmptrap(2)
SUN-ILOM-CONTROL-MIB::ilomCtrlAlertDestinationIP.1 = IPAddress: dest_ip_address
```

2. Start the trap receiver application, `snmptrapd`.
3. Generate a test trap to verify that traps are being sent by the agent (on the managed node) and received by the trap receiver (the management station).

While the daemon is running, log in to the Oracle ILOM CLI on the host that is running the SNMP agent and type the following command:

```
-> set /SP/alertmgmt/rules testalert=true
```

Note – It is important to test the trap daemon to make sure it is configured properly.

The following screen shows a sample output when a `testalert` trap is received at the management station:

```
SUN-ILOM-CONTROL-MIB::ilom.103.2.1.20.0 = STRING: "This is a test trap"
```


Index

A

- Active Directory, 38
 - Administrator Groups
 - viewing and configuring, 43
 - Alternate Server
 - viewing and configuring, 50
 - Custom Groups
 - viewing and configuring, 46
 - DNS Locator settings
 - viewing and configuring, 54
 - Operator Groups
 - view and configure, 44
 - User Domain
 - viewing and configuring, 48
- alert rules
 - CLI commands, 17
 - configuring, 82
- alerts
 - CLI commands for managing alerts, 17
 - generating email notification, 84

C

- clock settings
 - configuring network time protocol (NTP), 78
 - setting, 78
- component information
 - view, 76

E

- email alert settings
 - configuring, 86
- event log
 - configuring, 79

F

- firmware
 - viewing and configuring, 101

I

- IPMI
 - detailed specifications
 - location of, 120
 - functionality, 120
 - generating IPMI-specific traps, 120
 - IPMI Platform Event Trap (PET) alerts, 121
 - overview, 120
 - versions supported by ILOM, 120
- IPMItool
 - capabilities, 121
 - download site
 - location of, 121
 - functions of, 121
 - man page location, 121
 - references for, 121
 - running CLI commands with, 123
 - using IPMItool, 121

L

- LDAP, 58
 - configuring, 58
- LDAP/SSL, 61
 - Administrator Groups
 - MIB objects, 63
 - viewing and configuring, 62
 - Alternate Server
 - viewing and configuring, 68
 - certificate settings, 61
 - Custom Groups
 - viewing and configuring, 65
 - Operator Groups
 - viewing and configuring, 63
 - User Domain
 - viewing and configuring, 67

M

Management Information Base (MIB)
 definition, 3
 MIB tree, 3
 standard MIBs supported by ILOM, 5

N

Net-SNMP
 web site, 2

P

power consumption management
 entPhysicalName MIB object, 94
 monitoring individual power supply
 consumption using an snmpget command, 94
 monitoring permitted power
 snmpget command, 96
 monitoring power
 snmpget command, 93
 power monitoring
 snmpget command, 93
 sunPlatNumericSensor MIB objects, 94
 view and set power policy
 SNMP commands, 98, 106

R

RADIUS
 configuring, 71
redundancy settings
 view and configure, 53
remote Syslog receiver IP addresses
 configuring, 81

S

Single Sign On
 configuring, 37
 enabling, 37
single sign on
 overview, 37
SMTP clients
 configuring, 84
 MIB objects, 86
SNMP
 functions supported, 3
 managed node, 2
 management station monitoring, 2
 MIBs used to support ILOM, 5

Net-SNMP

 web site, 2
 network management station, 2
 tutorial web sites, 2
 versions supported, 2

SNMP traps

 configuring destinations using the web
 interface, 27

SNMP user accounts

 managing with the CLI, ?? to 17
 targets, properties, and values of, 11

SPARC boot mode, 117

SPARC diagnostics, 110

SPARC host settings, 113

SPARC key switch, 118

system alerts

 commands for managing, 17

T

Telemetry Harness Daemon (THD)

 configuring, 88

U

user accounts, 34